

Documentation 6.0

ZABBIX

07.06.2024

Contents

Manual do Zabbix	6
Copyright notice	6
1. Introdução	6
1 Estrutura do manual	6
2 O que é o Zabbix	7
3 Funcionalidades do Zabbix	7
4 Visão geral do Zabbix	8
5 O que há de novo no Zabbix 6.0.0	9
6 O que há de novo no Zabbix 6.0.1	20
6 O que há de novo no Zabbix 6.0.1	21
7 O que há de novo no Zabbix 6.0.2	21
8 O que há de novo no Zabbix 6.0.3	22
9 O que há de novo no Zabbix 6.0.4	22
10 O que há de novo no Zabbix 6.0.5	23
11 O que há de novo no Zabbix 6.0.6	24
12 O que há de novo no Zabbix 6.0.7	24
13 O que há de novo no Zabbix 6.0.8	25
14 O que há de novo no Zabbix 6.0.9	25
15 O que há de novo no Zabbix 6.0.10	25
16 O que há de novo no Zabbix 6.0.11	26
17 O que há de novo no Zabbix 6.0.12	26
18 O que há de novo na Zabbix 6.0.13	26
19 O que há de novo no Zabbix 6.0.14	28
20 O que há de novo no Zabbix 6.0.15	28
21 O que há de novo no Zabbix 6.0.16	29
22 O que há de novo no Zabbix 6.0.17	29
23 O que há de novo no Zabbix 6.0.18	29
25 O que há de novo no Zabbix 6.0.20	30
26 O que há de novo no Zabbix 6.0.21	30
27 O que há de novo no Zabbix 6.0.22	31
28 O que há de novo no Zabbix 6.0.23	32
29 O que há de novo no Zabbix 6.0.24	32
30 O que há de novo em Zabbix 6.0.25	33
31 O que há de novo no Zabbix 6.0.26	33
32 O que há de novo no Zabbix 6.0.27	34
33 O que há de novo no Zabbix 6.0.28	34
34 O que há de novo no Zabbix 6.0.29	35
35 O que há de novo no Zabbix 6.0.30	35
36 What's new in Zabbix 6.0.31	36
2. Definições	36
3. Processos Zabbix	38
2 Servidor	38
3 Agent 2	45
3 Agente	47
4 Proxy	50
5 Java gateway	52
6 Sender	56
7 Get	57
8 JS	57
9 Serviço Web	58

4. Instalação	58
1 Obtendo o Zabbix	58
2 Requisitos	59
3 Instalação a partir do código-fonte	71
4 Instalação a partir de pacotes	81
5 Instalação a partir de containers	96
6 Instalação da interface web	104
7 Atualizar procedimento	110
8 Problemas conhecidos	124
10 Notas de atualização para 6.0.0	131
11 Notas de atualização para 6.0.1	133
12 Notas de atualização para 6.0.2	133
13 Notas de atualização para 6.0.3	133
14 Notas de atualização para 6.0.4	134
15 Notas de atualização para 6.0.5	134
16 Notas de atualização para 6.0.6	134
17 Notas de atualização para 6.0.7	134
18 Notas de atualização para 6.0.8	134
19 Notas de atualização para 6.0.9	134
20 Notas atualizadas para 6.0.10	134
21 Notas atualizações para 6.0.11	135
22 Upgrade notas para 6.0.12	135
23 Notas de atualização para 6.0.13	135
24 Notas de upgrade para 6.0.14	136
25 Notas de atualização para 6.0.15	136
26 Notas de atualização para 6.0.16	136
27 Notas sobre a atualização para 6.0.17	136
28 Notas de atualização para 6.0.18	136
29 Upgrade notas para 6.0.19	136
30 Notas de atualização para 6.0.20	137
31 Notas de atualização para 6.0.21	137
32 Notas de atualização para 6.0.22	137
33 Notas de atualização para 6.0.23	137
34 Notas de atualização 6.0.24	138
35 Notas de atualização para 6.0.25	138
36 Notas de atualização para 6.0.26	138
37 Notas de atualização para 6.0.27	138
38 Notas atualizadas para 6.0.28	138
39 Atualizar notas para 6.0.29	138
39 Upgrade notes for 6.0.30	138
5. Início rápido	139
1 Autenticando e configurando usuário	139
2 Novo host	142
3 Novo item	144
4 Novo gatilho (trigger)	146
5 Recebendo notificação de problema	147
6 Novo modelo (template)	151
6. Aplicação Zabbix	153
7. Configuração	156
1 Hosts e grupos de host	164
2 Itens	172
3 Triggers	477
4 Eventos	494
5 Correlação de evento	497
6 Marcação	504
7 Visualização	507
8 Modelos	537
9 Modelos prontos para uso	538
10 Notificações sobre eventos	544
11 Macros	596
12 Usuários e grupos de usuários	607
13 Armazenamento de segredos	615
14 Relatórios programados	616

8. Monitoramento de serviço	620
1 Árvore de serviços	620
2 Ações de serviço	624
3 SLA	625
4 Exemplo Setup	626
9. Monitoramento web	632
1 Itens de monitoramento web	641
2 Cenário real de Monitoramento Web	643
10. Monitoramento de máquina virtual	651
1 Campos de chave de descoberta de máquina virtual	656
11. Manutenção	661
12. Expressões regulares	666
13. Reconhecimento do problema	672
14. Configuração exportação/importação	674
1 Grupos de anfitriões	676
2 Templates	676
3 Hosts	686
4 Mapas de rede	694
5 Tipos de mídia	700
15. Descoberta	706
1 descoberta de rede	706
2 Registro automático de agente ativo	715
3 Descoberta de baixo nível	718
16. Monitoração distribuída	771
Proxies	771
17. Criptografia	774
1 Usando certificados	780
2 Usando chaves pré-compartilhadas	787
3 Solução de problemas	789
18. Interface da Web	792
1 Menu	792
2 Seções de Frontend	793
3 Configurações do usuário	953
4 Pesquisa global	957
5 Modo de manutenção de front-end	959
6 Parâmetros da página	960
7 Definições	961
8 Criando seu próprio tema	962
9 Modo de depuração	962
10 Cookies usados pelo Zabbix	963
11 Fuso horário	963
13 Redefinindo senha	964
19 API	965
referência do método	970
Apêndice 1. Comentário de referência	1497
Apêndice 2. Mudanças de 5.4 para 6.0	1502
Mudanças na API do Zabbix na versão 6.0	1506
20. Módulos	1508
21. Apêndices	1514
1 Perguntas frequentes / Solução de problemas	1514
2 Instalação e configuração	1515
3 Configuração do processo	1557
4 Protocolos	1654
5 Itens	1680
6 Supported functions	1706
7 Macros	1786
8 Símbolos de unidade	1797
9 Sintaxe do período de tempo	1798
10 Execução de comandos	1799
11 Compatibilidade de versões	1800
14 Biblioteca Python para API do Zabbix	1800
14 Tratamento de erros do banco de dados	1801
14 Upgrade de monitoramento de serviço	1801

15 Biblioteca de links dinâmicos do remetente Zabbix para Windows	1802
18 Agente vs agente 2 comparação	1802
18 Exemplos de Escaping	1803
18 Outros problemas	1805

Páginas de manual do Zabbix	1806
zabbix_agent2	1806
NOME	1806
SINOPSE	1806
DESCRIÇÃO	1806
OPÇÕES	1807
ARQUIVOS	1807
VEJA TAMBÉM	1808
AUTOR	1808
Index	1808
zabbix_agentd	1808
NOME	1808
SINOPSE	1808
DESCRIÇÃO	1808
OPÇÕES	1808
ARQUIVOS	1809
VEJA TAMBÉM	1809
AUTOR	1809
Index	1809
zabbix_get	1810
NOME	1810
SINOPSE	1810
DESCRIÇÃO	1810
Opções	1810
EXEMPLOS	1811
VEJA TAMBÉM	1811
Índice	1811
Índice	1812
zabbix_js	1812
NOME	1812
SINOPSE	1812
DESCRIPTION	1812
OPÇÕES	1812
EXEMPLOS	1813
VEJA TAMBÉM	1813
Index	1813
zabbix_proxy	1813
NOME	1813
SINOPSE	1813
DESCRIÇÃO	1813
OPÇÕES	1813
ARQUIVOS	1814
VEJA TAMBÉM	1814
AUTOR	1814
Índice	1815
NOME	1815
SINOPSE	1815
DESCRIÇÃO	1815
OPÇÕES	1816
STATUS DE SAÍDA	1818
EXEMPLOS	1818
VEJA TAMBÉM	1818
Autor	1818
Index	1819
zabbix_server	1819
NOME	1819
SINOPSE	1819
DESCRIÇÃO	1819

OPÇÕES	1819
ARQUIVOS	1820
VEJA TAMBÉM	1820
AUTOR	1821
Index	1821
zabbix_web_service	1821
NOME	1821
SINOPSE	1821
DESCRIÇÃO	1821
OPÇÕES	1821
ARQUIVOS	1821
VEJA TAMBÉM	1822
AUTOR	1822
Índice	1822

Manual do Zabbix

Bem vindo ao Manual do usuário do software Zabbix. Estas páginas são criadas para ajudar os usuários a gerenciar com sucesso suas tarefas de monitoramento com o Zabbix, das simples às mais complexas.

Copyright notice

Zabbix documentation is NOT distributed under a GPL license. Use of Zabbix documentation is subject to the following terms:

You may create a printed copy of this documentation solely for your own personal use. Conversion to other formats is allowed as long as the actual content is not altered or edited in any way. You shall not publish or distribute this documentation in any form or on any media, except if you distribute the documentation in a manner similar to how Zabbix disseminates it (that is, electronically for download on a Zabbix web site) or on a USB or similar medium, provided however that the documentation is disseminated together with the software on the same medium. Any other use, such as any dissemination of printed copies or use of this documentation, in whole or in part, in another publication, requires the prior written consent from an authorized representative of Zabbix. Zabbix reserves any and all rights to this documentation not expressly granted above.

1. Introdução

Por favor utilize a barra lateral para acessar o conteúdo na seção de Introdução.

1 Estrutura do manual

Estrutura

O conteúdo deste manual é dividido em seções e subseções para fornecer fácil acesso a assuntos de interesse em particular.

Quando você navega até seções específicas, certifique-se de que você expanda as pastas da seção para revelar o conteúdo completo do que está incluso nas subseções e páginas individuais.

Associação cruzada entre páginas de conteúdo relacionado é fornecida tanto quanto possível de modo a garantir que informações relevantes não passem despercebidas pelos usuários.

Seções

Introdução fornece informações gerais sobre o software Zabbix atual. A leitura desta seção deve supri-lo com boas razões para escolher o Zabbix.

Conceitos do Zabbix explica a terminologia usada no Zabbix e provê detalhes sobre os seus componentes.

Instalação e Início rápido devem ajudá-lo a começar com o Zabbix. A seção **Zabbix appliance** é uma alternativa para ter uma prova rápida de como é usar o Zabbix.

Configuração é uma das maiores e mais importantes seções neste manual. Ela contém muita recomendação essencial sobre como configurar o Zabbix para monitorar seu ambiente, desde a configuração de hosts para obtenção de dados essenciais à visualização de dados, configuração de notificações e comandos remotos a serem executados em caso de problemas.

Serviços de TI detalha como usar o Zabbix para uma visão geral de alto nível do seu ambiente de monitoramento.

Monitoramento web deve ajudá-lo a aprender como monitorar a disponibilidade de sites web.

Monitoramento de máquina virtual apresenta como configurar o monitoramento de ambiente VMware.

Manutenção, Expressões regulares, Reconhecimento de evento e Exportação/Importação XML são seções adicionais que revelam como usar estes vários aspectos do software Zabbix.

Descoberta contém instruções para configuração de descoberta automática de dispositivos de rede, agentes ativos, sistemas de arquivo, interfaces de rede, etc.

Monitoramento distribuído lida com as possibilidades de uso do Zabbix em ambientes maiores e mais complexos.

Criptografia ajuda na explicação das possibilidades de criptografia de comunicações entre componentes do Zabbix.

Interface web contém informações específicas para uso da interface web do Zabbix.

API apresenta detalhes de operação com a API do Zabbix.

Listas detalhadas de informações técnicas estão inclusas nos **Apêndices**. Ali é onde você também encontrará uma seção de FAQ (perguntas frequentes).

2 O que é o Zabbix

Visão geral

O Zabbix foi criado por Alexei Vladishev, e atualmente é, ativamente, desenvolvido e suportado por Zabbix SIA.

O Zabbix é uma solução de nível enterprise, de código aberto e com suporte a monitoração distribuída.

O Zabbix é um software que monitora numerosos parâmetros de rede, a saúde e integridade de servidores, máquinas virtuais, aplicações, serviços, banco de dados, websites, a nuvem e muito mais. O Zabbix usa um mecanismo flexível de notificação que permite aos usuários configurar alertas baseados em e-mail para praticamente qualquer evento. Isso permite uma resposta rápida para problemas do servidor. O Zabbix oferece um excelente recurso de relatórios e visualização de dados baseados em dados armazenados. Isso torna o Zabbix ideal para gerenciamento de capacidade.

O Zabbix suporta tanto "pooling" quanto "trapping". Todos os relatórios e estatísticas, bem como os parâmetros de configuração são acessados através de um frontend baseado web. Um frontend baseado na web garante que o status de sua rede e a integridade de seus servidores podem ser avaliadas a partir de qualquer localização. Devidamente configurado, o Zabbix pode desempenhar um papel importante no monitoramento da infraestrutura de TI. Isto é igualmente verdadeiro para pequenas organizações com poucos servidores e para grandes empresas com milhares de servidores.

O Zabbix é gratuito. Zabbix é escrito e distribuído sob a GPL General Public License versão 2. Isso significa que seu código fonte é distribuído gratuitamente e disponível para o público em geral.

[Commercial support](#) está disponível e fornecido pela Zabbix Company e seus parceiros em todo o mundo.

Aprenda mais sobre [Zabbix features](#).

Usuários do Zabbix

Muitas organizações de diferentes tamanhos ao redor do mundo confiam no Zabbix como a plataforma principal de monitoramento.

3 Funcionalidades do Zabbix

Visão geral

O Zabbix é uma solução de monitoramento de rede altamente integrada, oferecendo uma variedade de funcionalidades em um único pacote.

Coleta de dados

- verificações de disponibilidade e performance
- suporte para monitoramento SNMP (ambos trapping e polling), IPMI, JMX, VMware
- verificações customizadas
- coleta de dados desejados em intervalos customizados
- executado pelo Server/Proxy e pelos agentes

Definições de limite (threshold) flexíveis

- você pode definir limites de problema muito flexíveis, chamados gatilhos, referenciando valores do banco de dados de backend

Alertas altamente configuráveis

- o envio de notificações pode ser customizado para o planejamento de escalação, destinatário, tipo de mídia
- notificações podem ser tornadas significantes e úteis usando variáveis de macro
- ações automáticas incluem comandos remotos

Gráfico em tempo real

- itens monitorados são imediatamente lançados em gráfico usando a funcionalidade nativa de criação de gráficos

Capacidades de monitoramento web

- O Zabbix pode seguir um caminho de cliques de mouse simulado em um site web e verificar pela funcionalidade e tempo de resposta

Opções de visualização extensivas

- habilidade de criar gráficos customizados que podem combinar múltiplos itens em uma única visualização
- mapas de rede

- apresentação em uma visão estilo dashboard
- relatórios
- visualização de alto nível (negócio) de recursos monitorados

Armazenamento de dados históricos

- dados armazenados em um banco de dados
- histórico configurável
- procedimento de limpeza (housekeeping) nativo

Configuração fácil

- adiciona dispositivos monitorados como hosts
- hosts são selecionados para monitoramento, uma vez que no banco de dados
- aplicar modelos ao dispositivos monitorados

Uso de modelos

- agrupamento de verificações em modelos
- modelos podem herdar outros modelos

Descoberta de rede

- descoberta automática de dispositivos de rede
- autoregistro de agente
- descoberta de sistemas de arquivo, interfaces de rede e OIDs SNMP

Interface web rápida

- um frontend baseado em web com PHP
- acessível de qualquer lugar
- você pode navegar com o mouse
- log de auditoria

API Zabbix

- a API fornece uma interface programável para o Zabbix para manipulações em massa, integração de softwares de terceiros e outros propósitos.

Sistema de permissões

- autenticação de usuário segura
- certos usuários podem ser limitados a certas visualizações

Agente com funcionalidade completa e facilmente extensível

- implementado nos alvos monitorados
- pode ser implementado em ambos Linux e Windows

Daemons binários

- escritos em C, para performance e pequena pegada de memória
- facilmente portátil

Pronto para ambientes complexos

- monitoramento remoto facilitado usando um Zabbix Proxy

4 Visão geral do Zabbix

Arquitetura

O Zabbix consiste de vários componentes de software importantes. Suas responsabilidades estão resumidas abaixo.

Server

O **Zabbix server** é o componente central para o qual os agentes reportam informações de disponibilidade e integridade e estatísticas. O Server é o repositório central no qual todas as configurações, estatísticas e dados operacionais são armazenados.

Armazenamento banco de dados

Todas as informações de configuração assim como os dados coletados pelo Zabbix são armazenados em um banco de dados.

Interface web

Para fácil acesso ao Zabbix de qualquer lugar e de qualquer plataforma, a interface baseada em web é oferecida. A interface é parte do Zabbix Server, e usualmente (mas não necessariamente) é executada na mesma máquina física na qual está sendo executado o Server.

Proxy

O **Zabbix proxy** pode coletar dados de performance e disponibilidade em nome do Zabbix Server. Um proxy é uma parte opcional da implementação do Zabbix; no entanto, ele pode ser de grande benefício para distribuir a carga de um único Zabbix Server.

Agent

Os Zabbix Agents são implementados nos alvos de monitoramento para monitorar ativamente os recursos locais e aplicações e disponibilizar os dados coletados para o Zabbix Server. Desde o Zabbix 4.4, há dois tipos de agente disponíveis: o **Zabbix Agent** (leve, suportado em muitas plataformas, escrito em C) e o **Zabbix Agent2** (extra-flexível, facilmente extensível com plugins, escrito em Go).

Fluxo de dados

Em adição é importante voltar um passo atrás e dar uma olhada geral no fluxo de dados dentro do Zabbix. De modo a criar um item que colete dados você deve primeiro criar um host. Indo para o outro lado do espectro do Zabbix você deve primeiro ter um item para criar um gatilho. Você deve ter um gatilho para criar uma ação. Portanto se você quiser receber um alerta de que sua carga de CPU está muito alta no Servidor X você deve primeiro criar uma entrada de host para o Servidor X seguido por um item para monitoramento de sua CPU, e então um gatilho que é ativado se a CPU estiver muito alta, seguida de uma ação que envia um e-mail para você. Ainda que isto possa parecer um monte de passos, com o uso de modelos realmente não é. No entanto, devido a este desenho é possível criar uma configuração muito flexível.

5 O que há de novo no Zabbix 6.0.0

Veja **breaking changes** para esta versão.

Cluster de alta disponibilidade para o servidor Zabbix

A nova versão vem com uma solução nativa de alta disponibilidade para o Servidor Zabbix.

A solução consiste em várias instâncias ou nós do `zabbix_server`, onde apenas um nó pode estar ativo (funcionando) por vez, enquanto outros nós estão em standby, prontos para assumir o controle caso o nó atual seja interrompido ou falhe ou falhar.

Veja também: **Cluster de alta disponibilidade**.

Monitoramento de serviços atualizado

Foram feitas várias atualizações no monitoramento de **serviços**. O monitoramento de serviços oferece uma visão de alto nível da infraestrutura monitorada no Zabbix.

**Mapeamento baseado em tags de serviços para problemas

A disponibilidade de **services** nas versões anteriores do Zabbix dependia de triggers e de seus estados. Na nova versão, isso foi substituído por um mapeamento baseado em tags para problemas o respectivo serviço.

Na configuração do serviço, as dependências rígidas e flexíveis não existem mais. Em vez disso, um serviço pode ter vários serviços principais.

Menu de serviços

Agora há um novo menu Serviços no Zabbix, com quatro seções de menu:

- **Services** - para visão geral e configuração do serviço (movido de Monitoramento -> Serviços)
- **Service actions** - para ações de serviço (novo tipo de ação) tipo)
- **SLA** - para configurar SLAs
- **SLA report** - para relatórios de SLA (também disponível como widget de painel)

Monitoring	
Services	
Services	
Service actions	
SLA	
SLA report	

Name	Status
Availability 2	High
Disc space	OK
Example service	OK

Observe que não existe mais uma seção separada para a configuração do serviço (anteriormente em Configuração → Serviços).

****Regras de cálculo e propagação de status**

Há novas regras de cálculo de status e regras adicionais flexíveis para calcular o status de um serviço pai com base nos status e no peso dos filhos diretos. Agora também é possível definir regras flexíveis para propagar o status de um serviço status de serviço para serviços pai.

Permissões

As permissões flexíveis para serviços foram implementadas em **função do usuário** nível. O acesso de leitura-escrita ou somente leitura pode ser concedido a todos, a nenhum ou a serviços selecionados (com base no nome ou nas tags).

****Análise da causa raiz**

Uma nova coluna Causa raiz lista os problemas subjacentes que afetam direta ou indiretamente o status do serviço.

Services

Name	Status	Root cause
Availability 2	High	Nodata trigger, Nodata trigger 1h

Se você clicar no nome do problema, poderá ver mais detalhes sobre ele em Monitoramento → Problemas.

****Alerta sobre mudança de status do serviço**

Agora é possível receber alertas automatizados sobre alterações no status do serviço, semelhantes aos alertas sobre alterações no status do trigger mudanças.

Uma nova funcionalidade **service action** foi adicionada, semelhante a outras ações no Zabbix. As ações de serviço podem incluir etapas para operações de problema, recuperação e atualização relacionadas a serviços. É possível configurar dois tipos de ações: enviar uma mensagem para os destinatários especificados e executar um comando remoto no servidor Zabbix no servidor Zabbix. Da mesma forma que as ações de acionamento, as ações de serviço suportam problemas **escalonamento**.

Novos modelos de mensagem Service, Service recovery e Service update foram adicionados a **media types** e devem ser definidos para permitir o envio correto de notificações para ações de serviço.

****Clonagem de serviços**

Os serviços agora podem ser clonados. O botão Clone foi adicionado ao **formulário de configuração** de um serviço. Quando um serviço é clonado, seus links pai são preservados, enquanto os links filho não são.

Chaves primárias

As chaves primárias agora são usadas em todas as tabelas, inclusive nas tabelas de histórico, em novas instalações.

Não há atualização automática para chaves primárias nas instalações existentes. As instruções para uma **atualização manual** das tabelas de histórico para chaves primárias em instalações pré-existentes estão disponíveis em **MySQL/MariaDB**, **PostgreSQL**, **TimescaleDB v1** e **v2**, e **Oracle**.

Widgets Vários widgets de painel foram adicionados na nova versão.

Principais hosts

Um widget Top hosts foi adicionado aos widgets do painel. Esse widget foi projetado para substituir o widget Data overview (Visão geral dos dados) que agora está obsoleto.

O widget Top hosts permite criar tabelas personalizadas para a visão geral dos dados, o que é útil para relatórios do tipo Top N e relatórios de progresso de barras úteis para o planejamento de capacidade.

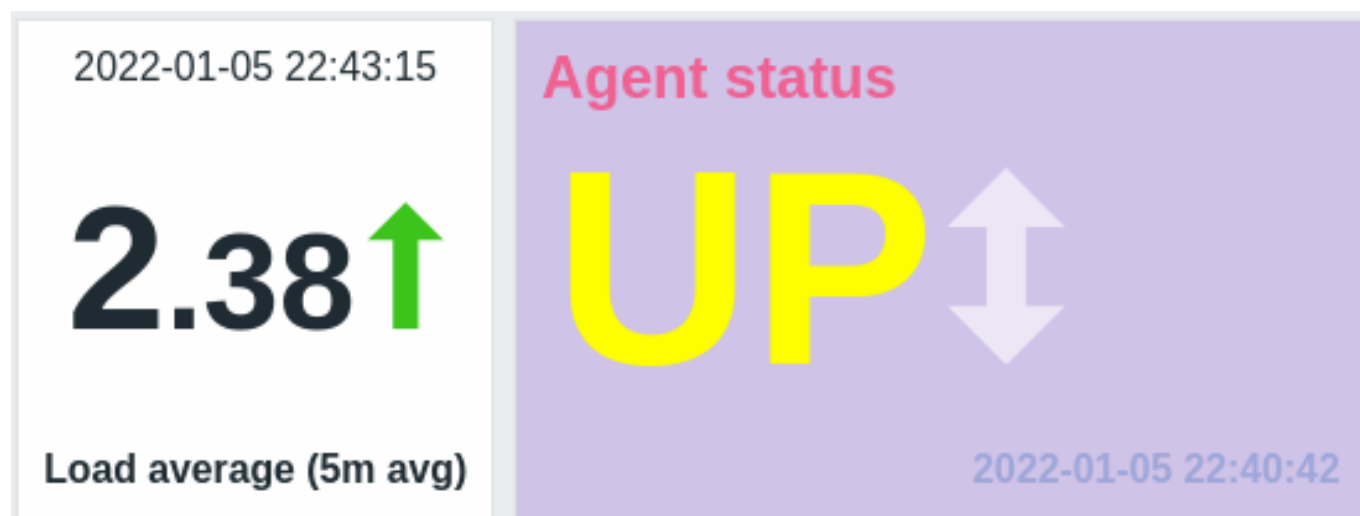
Top 3		
Host	CPU	
Server3		3.16
Zabbix server		3.11
New host		3.1

Para obter mais informações, consulte [Widget Top hosts](#).

Valor do item

Um widget Item value foi adicionado aos widgets do painel.

Esse tipo de widget é útil para exibir valores de itens individuais com destaque. São possíveis diferentes estilos visuais de exibição possíveis:



Para obter mais informações, consulte [Item value widget](#).

Macros

Novas macros

Agora há suporte para novas macros para depuração de expressões de acionamento e ações internas.

As macros de depuração de expressões simplificam o processo de depuração de expressões de acionamento:

- {TRIGGER.EXPRESSION.EXPLAIN}, {TRIGGER.EXPRESSION.RECOVERY.EXPLAIN} - resolvem para uma expressão de acionamento ou recuperação parcialmente avaliada ou expressão de recuperação, em que somente as funções são aplicadas;
- {FUNCTION.VALUE<1-9>}, {FUNCTION.RECOVERY.VALUE<1-9>} - resolvem para os resultados da enésima função baseada em item no momento do evento.

As macros para ações internas contêm o motivo pelo qual um item, uma regra LLD ou um trigger deixou de ser suportado:

- {ITEM.STATE.ERROR} - para notificações internas baseadas em itens;
- {LLDRULE.STATE.ERROR} - para notificações internas baseadas em regras LLD;
- {TRIGGER.STATE.ERROR} - para notificações internas baseadas em triggers.

Para obter mais detalhes, consulte [Macros Suportadas](#).

Macros simples substituídas por macros de expressão

Uma nova sintaxe de expressão para triggers e itens calculados foi introduzida em [Zabbix 5.4] (<https://www.zabbix.com/documentation/5.4/m>). No entanto, a sintaxe antiga ainda permaneceu em uso nas macros simples. Na nova versão, a funcionalidade das macros simples foi transferida para as macros de expressão e a nova sintaxe de expressão é usada. Consulte a comparação abaixo para obter detalhes sobre a alteração:

No Zabbix 6.0	Antes do Zabbix 6.0
<code>{?avg(/host/key,1h)}</code>	<code>{host:key.avg(1h)}</code>
Exemplo de uma macro de expressão na nova versão.	Exemplo de uma macro simples nas versões anteriores.

As macros simples existentes serão convertidas em macros de expressão durante a atualização. O escopo das macros de expressão abrange o mesmo que era oferecido pelas macros simples. Portanto, as macros de expressão podem ser usadas em:

- notificações e comandos de problemas
- notificações e comandos de atualização de problemas
- rótulos de elementos do mapa
- rótulos de links de mapas
- rótulos de formas de mapas
- nomes de gráficos

Macros posicionais não são mais suportadas

O suporte a macros posicionais no nome do item (\$1, \$2...\$9), obsoleto desde o Zabbix 4.0, foi totalmente removido.

Não há mais suporte para macros de usuário no nome do item

O suporte a macros de usuário em nomes de itens (incluindo nomes de regras de descoberta), obsoleto desde o Zabbix 4.0, foi totalmente removido.

Processamento em massa para métricas do Prometheus

O processamento em massa de itens dependentes foi introduzido na fila de pré-processamento para melhorar o desempenho da recuperação de métricas do Prometheus.

Consulte [Prometheus checks](#) para obter mais detalhes.

Processamento de resultados para o padrão Prometheus

Uma etapa de padrão do Prometheus no pré-processamento pode produzir um resultado em que várias linhas são correspondidas. Para lidar com essa situação, um novo [parâmetro](#) no processamento de resultados foi adicionado à etapa de pré-processamento do padrão Prometheus que permite agregar os dados de várias linhas correspondentes, introduzindo funções como soma, mínimo, máximo, média e contagem.

Funções Funções para histogramas do Prometheus

Há algum tempo é possível coletar [métricas do Prometheus](#) no Zabbix mas é difícil trabalhar com algumas das métricas. Especificamente, as métricas do tipo histograma podem ser apresentadas no Zabbix como vários itens com os mesmos nomes de chave, mas com parâmetros diferentes. No entanto, mesmo que esses itens estejam logicamente relacionados e representem os mesmos dados, tem sido difícil analisar os dados coletados sem funções especializadas. Para cobrir essa lacuna de funcionalidade na nova versão, as funções **rate()** e **histogram_quantile()** que produzem o mesmo resultado que suas contrapartes PromQL, foram adicionadas.

Outras novas adições para complementar essa funcionalidade são as funções **bucket_rate_foreach()** e as funções **bucket_percentile()**. Para obter mais informações, consulte:

- [Funções de histórico](#) (consulte **rate()**)
- [Funções de agregação](#) (consulte **histograma_quantile()**, **bucket_percentile()**)
- [Funções Foreach](#) (consulte **bucket_rate_foreach()**)

Mudança monotônica

Agora é possível verificar se há aumento ou diminuição monotônica nos valores dos itens usando as novas **monoinc()** ou **monodec()** [funções de histórico](#).

Contagem de alterações

Uma nova **função de histórico** **changecount()** foi adicionada, permitindo contar o número de alterações entre valores adjacentes. A função suporta três modos diferentes para contar todas as alterações, somente diminuições ou somente aumentos. Por exemplo, ela pode ser usada para rastrear alterações no número de usuários ou no número de diminuições no tempo de atividade do sistema.

Contagem de entidades

Novas **funções** foram adicionadas para simplificar a contagem de hosts, itens ou valores específicos, retornados por **funções** **foreach**.

Funções de agregação:

- **count** - número total de valores em uma matriz retornada por uma função **foreach** (retorna um número inteiro);
- **item_count** - número total de itens atualmente habilitados que correspondem aos critérios de filtro (retorna um número inteiro).

Função **foreach**:

- **exists_foreach** - número de itens ativados no momento que correspondem aos critérios de filtro (retorna uma matriz).

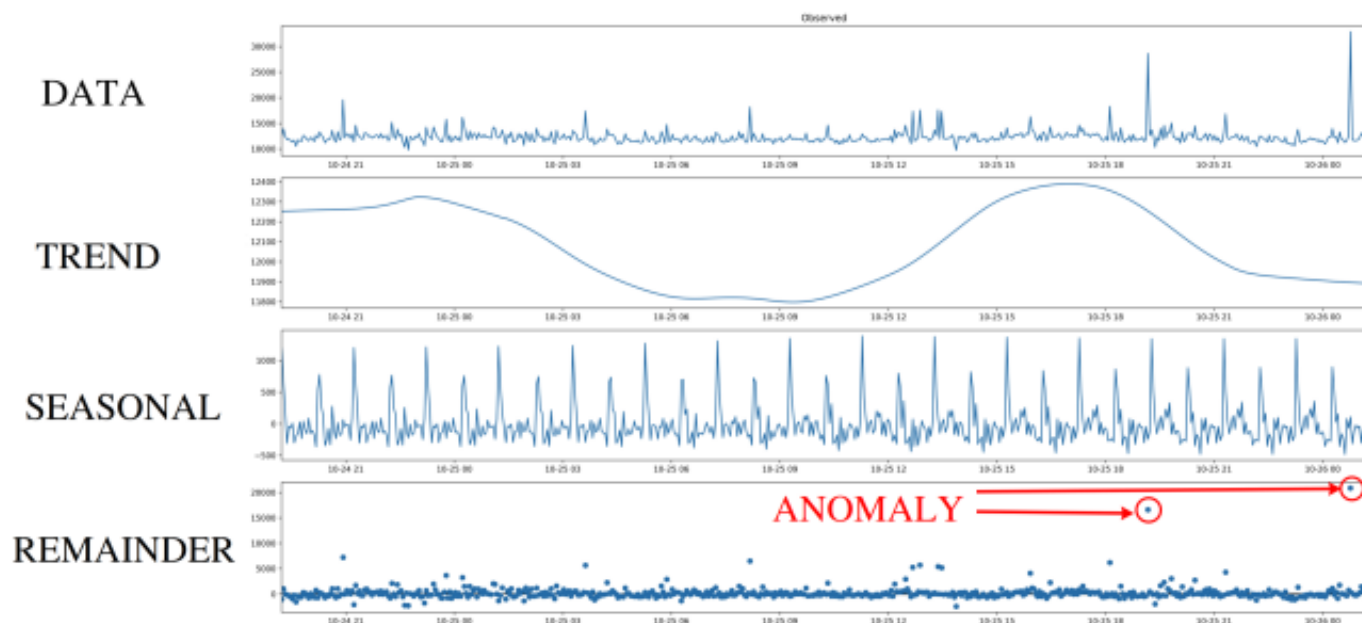
Deteção de anomalias

O Zabbix 5.2 introduziu novas funções de tendência úteis para o monitoramento da linha de base. No entanto, elas ainda exigem a definição de limites (por exemplo, verificar se o tráfego da Web em setembro de 2021 é menos e 2x maior em comparação com setembro de 2020). Existem casos de uso em que esses limites são difíceis de definir. Por exemplo, o tráfego da Web de um site novo, mas altamente popular pode crescer organicamente muitas vezes em um ano, mas a taxa de crescimento é desconhecida. No entanto, um pico repentino de tráfego devido a um ataque DDOS deve gerar um alerta independentemente do crescimento orgânico do tráfego.

Os algoritmos de detecção de anomalias fazem exatamente isso: encontram dados que não parecem normais (outliers) em um contexto de outros valores.

Foi adicionada uma nova **função de tendência** **trendstl()** que usa o método de "decomposição para calcular a taxa de anomalia. Ela divide uma única sequência de série temporal em três outras sequências:

- sequência de tendências que contém apenas grandes alterações nos dados originais (por exemplo, o tráfego do site mostra crescimento)
- sequência de estação que contém apenas alterações sazonais (por exemplo, menos tráfego no site no verão e mais no outono)
- sequência restante, que contém apenas valores residuais que não podem ser interpretados como partes da tendência ou da estação



A detecção de anomalias funciona com a sequência restante e verifica se há valores que estão muito distantes da maioria dos valores restantes. "Longe" significa que o valor absoluto da sequência restante é N vezes maior do que o desvio padrão ou médio ou desvio médio.

Funções String

String function **concat** agora permite concatenar mais de dois parâmetros. Essa função pode ser usada para combinar cadeias de caracteres e valores em diferentes combinações ou anexar dois ou mais valores uns aos outros. Os tipos de dados também são compatíveis.

Itens Seleção automatizada de tipos

O formulário de configuração do item agora sugere automaticamente o tipo de informação correspondente, se a chave do item selecionado retornar dados apenas do tipo específico (por exemplo, o item **log[]** requer Type of information: Log). O parâmetro Type of information agora está localizado sob o parâmetro Key na guia Item principal e é duplicado na guia na guia Preprocessing se pelo menos uma etapa de pré-processamento for especificada. Se o Zabbix detectar uma possível incompatibilidade entre o tipo de informação e chave selecionados, um ícone de aviso será exibido ao lado do campo Tipo de informação.

Itens do agente

Vários **novos** itens foram adicionados ao agente/agente 2 do Zabbix:

- **agent.hostmetadata** - retorna metadados do host
- **kernel.openfiles** - retorna o número de descritores de arquivos abertos
- **net.tcp.socket.count[]** - retorna o número de soquetes TCP que correspondem aos parâmetros
- **net.udp.socket.count[]** - retorna o número de soquetes UDP que correspondem aos parâmetros
- **vfs.dir.get[]** - retorna a lista de arquivos de diretório como JSON
- **vfs.file.get[]** - retorna informações sobre um arquivo como JSON
- **vfs.file.owner[]** - retorna a propriedade de um arquivo
- **vfs.file.permissions[]** - retorna uma cadeia de 4 dígitos contendo número octal com permissões Unix

Além disso:

- **vfs.file.cksum[]** agora suporta um segundo parâmetro mode (crc32, md5, sha256)
- **vfs.file.size[]** agora oferece suporte a um segundo parâmetro mode (bytes ou lines)
- **vfs.fs.discovery** e **vfs.fs.get** agora retornam uma macro {#FSLABEL} no Windows (com nomes de volumes)

Para obter mais detalhes, consulte [agent items](#).

Itens calculados

Os itens calculados agora suportam não apenas informações numéricas, mas também dos tipos **texto**, **log** e **caracteres**.

Recarregamento de parâmetros do usuário sem reinicialização do agente

Os parâmetros do usuário agora podem ser recarregados do arquivo de configuração sem reiniciar o agente. Para fazer isso, execute a nova opção de controle de tempo de execução `userparameter_reload` opção de controle de tempo de execução, por exemplo:

```
zabbix_agentd -R userparameter_reload
```

ou

```
zabbix_agent2 -R userparameter_reload
```

UserParameter é a única opção de configuração do agente que será recarregada com esse comando.

Controles de tempo de execução em sistemas operacionais baseados em BSD

Anteriormente, as opções de controle de tempo de execução do Zabbix Server e do Zabbix Proxy não eram suportadas em sistemas baseados em BSD. A alteração do método de transferência de comandos em tempo de execução permitiu eliminar essa limitação. Agora, a maioria dos comandos é compatível com em FreeBSD, NetBSD, OpenBSD e outros sistemas operacionais da família ***BSD**.

Para obter a lista exata, consulte Controle de tempo de execução para Zabbix [server](#) ou [proxy](#).

Plug-ins do Zabbix Agent 2

****Carregador de plug-in externo**

Anteriormente, os plug-ins só podiam ser compilados no Zabbix Agent 2, o que exigia a recompilação do agente toda vez que você precisasse alterar o conjunto de plug-ins disponíveis. Agora, com a adição do carregador de plug-ins externo, os plug-ins não precisam ser integrados diretamente ao agente 2 e podem ser adicionados como complementos externos separados (plug-ins carregáveis), facilitando assim o processo de criação de plug-ins adicionais para coletar novas métricas de monitoramento.

A introdução de plug-ins carregáveis causou as seguintes alterações nos parâmetros de configuração:

- o parâmetro `Plugins.<PluginName>.Path` foi movido para `Plugins.<PluginName>.System.Path`.
- o parâmetro `Plugins.<PluginName>.Capacity`, embora ainda suportado, foi descontinuado; use `Plugins.<PluginName>.System.Capacity`.

Requisitos de senha

Os requisitos personalizados de complexidade de senha agora podem ser fornecidos para o Zabbix interno (/manual/web_interface/frontend_se). Para evitar que os usuários do Zabbix definam senhas fracas, é possível impor as seguintes restrições:

- Definir o comprimento mínimo da senha.

- Exigir que a senha contenha uma combinação de letras maiúsculas e letras maiúsculas e minúsculas, dígitos e/ou caracteres especiais.
- Proibir o uso das senhas mais comuns e fáceis de adivinhar.

Bancos de dados Para criar a melhor experiência para o usuário e garantir o melhor desempenho do Zabbix em vários ambientes de produção, o suporte a algumas versões mais antigas de bancos de dados foi abandonado. Isso se aplica principalmente às versões de banco de dados que estão chegando ao fim de sua vida útil e versões com problemas não corrigidos que podem interferir no desempenho normal.

A partir do Zabbix 6.0, as seguintes versões de **banco de dados** são oficialmente suportadas:

- MySQL/Percona 8.0.X
- MariaDB 10.5.X - 10.6.X
- PostgreSQL 13.X - 14.X
- Oracle 19c - 21c
- TimescaleDB 2.0.1-2.3
- SQLite 3.3.5-3.34.X

Por padrão, o Zabbix Server e o proxy não serão iniciados se uma versão de banco de dados não suportada for detectada. Agora é possível, embora não seja recomendado, agora é possível desativar a verificação da versão do banco de dados modificando o parâmetro de configuração AllowUnsupportedDBVersions do o **server** ou **proxy**.

suporte utf8mb4 para MySQL

a codificação utf8mb4 com agrupamento utf8mb4_bin agora é suportada para instalações do Zabbix com o banco de dados MySQL/MariaDB.

Anteriormente, apenas a codificação utf8 era suportada, o que no MySQL significa codificação utf8mb3 e, portanto, suporta apenas um subconjunto de caracteres UTF-8 adequados. Na nova versão, o suporte a utf8mb4 foi adicionado com suporte para **full** Conjunto de caracteres UTF-8. As instalações antigas que usam utf8mb3 são mantidas intactas e podem continuar usando essa codificação.

Consulte também as instruções sobre a execução da **conversão utf8mb4** após a atualização para a versão 6.0.

Processos Tempo limite do Zabbix get e do Zabbix sender

Os utilitários Zabbix get e Zabbix sender agora suportam um parâmetro `-t <seconds>` ou `--timeout <seconds>` de tempo limite. O intervalo válido é:

- 1-30 segundos para o Zabbix get (padrão: 30 segundos)
- 1-300 segundos para o Zabbix sender (padrão: 60 segundos)

Funcionalidade estendida do gateway SNMP

O gateway SNMP agora pode fornecer informações sobre triggeres em um estado problemático e revelar informações sobre o host nos detalhes do trigger

Além disso, agora é possível limitar a taxa de traps de SNMP enviados pelo gateway de SNMP.

A lista de OIDs suportados foi ampliada com um novo OID **.10** para uma lista delimitada por vírgulas de nomes de host de triggeres.

Novos parâmetros foram adicionados ao arquivo de configuração do gateway SNMP:

- ProblemBaseOID - OID da tabela de triggeres de problemas;
- ProblemMinSeverity - gravidade mínima, os triggeres com gravidade inferior não serão incluídos;
- ProblemHideAck - se especificado, somente os triggeres com problemas não reconhecidos serão incluídos;
- ProblemTagFilter - se especificado, somente os triggeres com o nome de tag especificado serão incluídos;
- TrapTimer - se definido, o Zabbix não enviará mais do que um trap de maior severidade em um determinado período de tempo.

Para obter detalhes, consulte [Zabbix SNMP Gateway] (<https://git.zabbix.com/projects/ZT/repos/snmp-gateway/browse/README.md>).

Conteúdo compactado no monitoramento da Web

A capacidade de lidar com conteúdo compactado foi adicionada ao Zabbix Web do Zabbix. Todos os formatos de codificação suportados por **libcurl** são suportados.

Pré-processamento Linguagem de consulta do Prometheus

O pré-processamento do Zabbix Prometheus **query language** agora suporta dois operadores adicionais de correspondência de rótulos:

- **!=** -- seleciona rótulos que não são iguais à string fornecida;

- `!~` -- seleciona rótulos que não correspondem à cadeia de caracteres fornecida string fornecida.

Métodos JavaScript

Os métodos HTTP PATCH, HEAD, OPTIONS, TRACE e CONNECT foram adicionados ao mecanismo JavaScript. Além disso, o mecanismo agora permite enviar solicitações de métodos HTTP personalizados com o novo método JS `HttpRequest.customRequest`.

Veja também: [Objetos JavaScript adicionais](#).

Registro de auditoria

**Registros

O log de auditoria agora contém registros sobre todas as alterações de configuração para todos os objetos do Zabbix, incluindo alterações que ocorreram como resultado da execução de uma regra LLD, uma ação de descoberta de rede, uma ação de autorregistro ou uma execução de script. Anteriormente, as alterações de configuração iniciadas a partir do Zabbix Server, por exemplo, como resultado da execução de uma regra de descoberta, não eram registradas. Agora, essas modificações de objeto serão armazenadas como registros de auditoria atribuídos ao usuário System.

**Filtro de registro

Foi adicionada uma funcionalidade para filtrar registros pela operação de front-end que causou essas entradas. Se vários registros de log tiverem sido criados como resultado de uma única operação, por exemplo, vincular/desvincular um modelo, esses registros terão o mesmo Recordset ID.

**Configurações de auditoria

Novo [seção](#) Audit log foi adicionada ao menu Administration→General, permitindo ativar ou desativar o registro de auditoria. As configurações de manutenção para auditoria, anteriormente localizadas na seção Housekeeper, também foram movidas para a nova seção Log de auditoria.

Suporte ao PCRE2

O suporte ao PCRE2 foi adicionado e os pacotes de instalação do Zabbix para RHEL 7 e mais recentes, SLES (todas as versões), Debian 9 e mais recentes, Ubuntu 16.04 e mais recentes foram atualizados para usar o PCRE2. O PCRE ainda é suportado, mas o Zabbix só pode ser compilado com uma das bibliotecas PCRE ou PCRE2, ambas não podem ser usadas ao mesmo tempo ao mesmo tempo.

Processamento separado para verificações de ODBC

O processamento de verificações de ODBC foi transferido dos processos regulares de sondagem para processos separados de servidor/proxy sondadores de ODBC. Essa alteração permite limitar o número de conexões ao banco de dados criadas pelos processos de sondagem. Anteriormente, as verificações de ODBC eram executadas por pollers regulares, que também trabalham com itens do agente Zabbix, verificações SSH etc.

Um novo parâmetro de configuração `StartODBCPollers` foi adicionado ao Zabbix nos arquivos de configuração do Zabbix [server](#) e [proxy](#).

Você pode usar o item interno `zabbix[process,<type>]` para monitorar a carga dos pollers ODBC.

Integrações de webhooks

Está disponível uma nova integração que permite usar o tipo de mídia [webhook](#) para criar [Github issues](#) das notificações do Zabbix.

Modelos (Templates) Novos modelos oficiais estão disponíveis para monitoramento.

Kubernetes

- *Nós do Kubernetes por HTTP
- *Estado do cluster do Kubernetes por HTTP
- *Servidor de API do Kubernetes por HTTP
- *Gerenciador de controle do Kubernetes por HTTP
- *Agendador do Kubernetes por HTTP
- *Kubelet do Kubernetes por HTTP

Para ativar o monitoramento do Kubernetes, você precisa usar a nova ferramenta [Zabbix Helm Chart](#), que instala o proxy Zabbix e os agentes Zabbix no cluster do Kubernetes.

Para saber mais sobre a configuração de modelos, consulte [HTTP template operation](#).

Mikrotik

- MikroTik <device model> SNMP - 53 novos modelos específicos para monitorar vários modelos de roteadores e switches MikroTik roteadores e switches ethernet, consulte [lista completa] (<https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/>)

- Mikrotik SNMP - um modelo genérico para monitorar dispositivos MikroTik.

Você pode obter esses modelos:

- Em Configuration → Templates em novas instalações;
- Ao atualizar a partir de versões anteriores, os modelos mais recentes podem ser baixados do [Zabbix Git repository] (<https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates>) e importados manualmente para o Zabbix na seção Configuration → Templates. Se já existir um modelo com o mesmo nome, verifique a opção Delete missing antes de importar para obter uma importação limpa. Dessa forma, os itens que foram excluídos do modelo atualizado serão removidos (observe que o histórico dos itens excluídos será perdido)

Notificações Links de modelos mais visíveis

Para tornar a vinculação de modelos mais visível, agora ela é colocada na primeira guia dos formulários de configuração de host, protótipo de host e modelo e nos formulários de atualização em massa de host/template.

Host	IPMI	Tags	Macros 2	Inventory ●	Encryption	Value mapping 1												
<p>* Host name <input type="text" value="Zabbix server"/></p> <p>Visible name <input type="text" value="Zabbix server"/></p> <table border="1"> <thead> <tr> <th>Templates</th> <th>Name</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td></td> <td>Linux OS agent</td> <td>Unlink Unlink and clear</td> </tr> <tr> <td></td> <td>App Zabbix Server</td> <td>Unlink Unlink and clear</td> </tr> <tr> <td colspan="3"> <input type="text" value="type here to search"/> </td> </tr> </tbody> </table>							Templates	Name	Action		Linux OS agent	Unlink Unlink and clear		App Zabbix Server	Unlink Unlink and clear	<input type="text" value="type here to search"/>		
Templates	Name	Action																
	Linux OS agent	Unlink Unlink and clear																
	App Zabbix Server	Unlink Unlink and clear																
<input type="text" value="type here to search"/>																		

Consequentemente, uma guia separada para vinculação de modelos foi removida de todos os respectivos formulários.

Em um desenvolvimento relacionado, na configuração do protótipo de host, os campos para a seleção do protótipo do grupo de host/grupo de host também foram movidos de uma guia separada para uma guia separada também foram movidos de uma guia separada para a primeira guia.

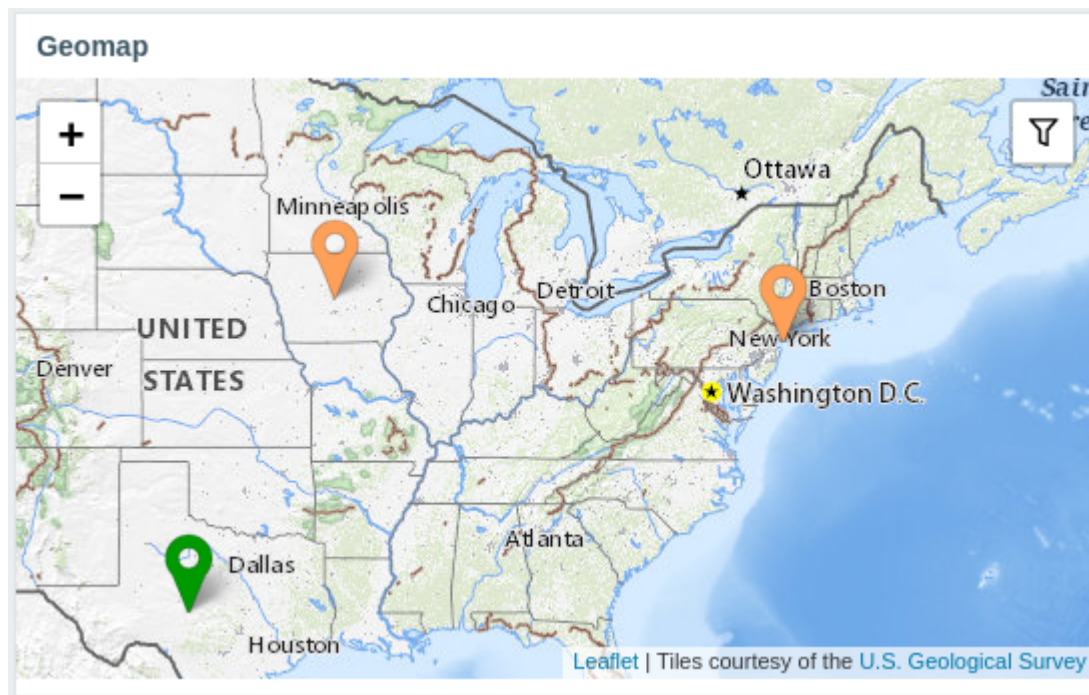
Transferência de comandos em tempo de execução

Os comandos de tempo de execução do Zabbix Server e do proxy agora são enviados via socket em vez de sinais Unix. Essa alteração permitiu melhorar a experiência do usuário ao trabalhar com opções de controle de tempo de execução:

- Os resultados da execução do comando agora são impressos no console.
- É possível enviar parâmetros de entrada mais longos, como o nome do nó HA em vez do número do nó.

Front-end Geomap

Um novo widget de mapa geográfico para os painéis foi introduzido, oferecendo uma maneira de exibir hosts em mapas geográficos. Para obter mais informações, consulte o Geomap [dashboard widget](#) e [mapas geográficos](#).



Subfiltro nos dados mais recentes

Um subfiltro foi adicionado à seção Latest data. O subfiltro é útil para o acesso rápido, com um clique, a grupos de itens relacionados.

O subfiltro mostra **links clicáveis** que permitem filtrar itens com base em uma entidade comum: o host, o nome da tag ou o valor da tag. Assim que a entidade é clicada, os itens são imediatamente filtrados.

Para obter mais detalhes, consulte a seção [latest data](#).

Melhorias na usabilidade de gráficos personalizados

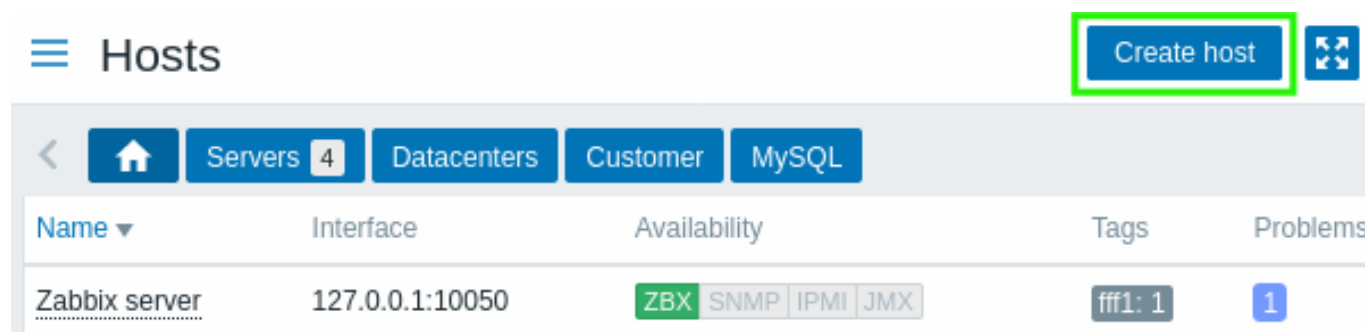
A página de gráficos em Monitoring → Hosts → Graphs recebeu vários aprimoramentos de usabilidade:

- Não há mais um limite de 20 gráficos na página
- Foi adicionado um subfiltro que permite selecionar rapidamente grupos de gráficos relacionados com base em uma tag ou valor de tag comum
- Gráficos simples para o host podem ser exibidos juntamente com gráficos personalizados

Para obter mais detalhes, consulte a página [graph](#).

Criação de hosts a partir do monitoramento

Agora também é possível criar novos hosts a partir de Monitoramento → [Hosts](#).



O botão Create host está disponível para usuários Admin e Super Admin.

Edição de host como pop-up

O formulário para criação e edição de host agora é aberto em uma janela modal (pop-up) em Configuration → Hosts, Monitoring → Hosts e em qualquer página, onde houver um menu de host ou outro link direto para a configuração do host configuração.

Os links diretos para a página de edição do host ainda funcionam e estão abrindo a página de edição do host em página inteira página de edição do host em página inteira.

Melhor navegação entre a configuração do item e os dados mais recentes

Um novo menu de contexto para itens foi introduzido em **Latest data**, permitindo acessar a configuração do item e os gráficos disponíveis configuração do item e os gráficos disponíveis:

<input type="checkbox"/> Host	Name ▲	Last check
<input type="checkbox"/> Zabbix server	/: Free inodes in %	5s
<input type="checkbox"/> Zabbix server	/: Free space ?	33s
<input type="checkbox"/> Zabbix server	/: Sp	6s
<input type="checkbox"/> Zabbix server	/: Tot	7s
<input type="checkbox"/> Zabbix server	/: Us	8s
<input type="checkbox"/> Zabbix server	A Int	19s

Por outro lado, um novo menu de contexto foi introduzido na seção **lista de itens** no menu de configuração que permite acessar os dados mais recentes do item e outras opções úteis:

Items

All hosts / Zabbix server Enabled ZBX SNMP IPMI Items 146 Triggers 67 Graphs 27 Discover

<input type="checkbox"/>	Name
<input type="checkbox"/>	Template Module Linux generic by Zabbix agent: Maximum number of open file descriptors

0 selected

ear history Conv Mass update

Configured max number of open filedescript

Esse menu substitui a opção do assistente nas versões anteriores. Um menu semelhante foi também introduzido para **itens de modelo** e **item prototypes**.

Notificação sobre escalonamentos cancelados

Ao configurar [operações de ação] (/manual/config/notifications/action/operation#configuring-an-operation), agora é possível possível cancelar as notificações sobre escalonamentos cancelados desmarcando a caixa de seleção da opção correspondente.

Monitoring → Dados mais recentes atualizados

Foram feitos vários aprimoramentos na seção Dados mais recentes:

- O tempo desde a última verificação (por exemplo, 1m 20s) agora é exibido em vez do tempo de execução do último item.
- Passar o mouse sobre o último valor de um item mostrará o valor bruto sem unidades ou mapeamento de valor aplicado.
- Se um host estiver em manutenção, um ícone de chave inglesa laranja ficará visível ao lado do nome do host.

Monitoramento → Visão geral removida

A seção Visão geral no menu Monitoramento foi completamente removida. A mesma funcionalidade ainda pode ser acessada usando os painéis Data overview e Trigger overview dashboard **widgets**.

Diversos

- O tamanho máximo do campo foi aumentado para os seguintes campos:
 - **Item preprocessing parameters**

- **Tipo de mídia** mensagem
- O idioma padrão da interface web do Zabbix foi alterado de Inglês britânico para inglês americano. O suporte ao inglês britânico foi abandonado.
- O link Share no menu principal foi substituído por um link Integrations que leva à página [Integrações](#) no site da Site do Zabbix.
- Se a interface web do Zabbix for aberta em um dos idiomas disponíveis no site do Zabbix, ao clicar no link Integrations, o usuário poderá acessar a página Integrations no site do Zabbix disponíveis no site do Zabbix, ao clicar no link Integrations, será aberta a página Integrations no idioma apropriado. Para todos os outros idiomas, incluindo o inglês, a página de Integrações será aberta em inglês.
- Uma expressão personalizada, usada em **action configuration** para calcular as condições, agora pode ter até 1024 caracteres (antes era 255).
- A seção Monitoramento->Hosts agora mostra o link para a tela de problemas do host, mesmo que nenhum problema esteja aberto no momento.

Mudanças significativas Registro de auditoria

Para implementar as alterações na [funcionalidade de registro de auditoria] (#audit-log), a estrutura de banco de dados existente anteriormente teve de ser reformulada. Durante a atualização, as tabelas de banco de dados auditlog e auditlog_details serão substituídas pela nova tabela auditlog com um formato diferente. **Os registros de log de auditoria existentes serão excluídos

Verificação de versões de BD compatíveis

Zabbix **server** e **proxy** agora verificarão a versão do banco de dados versão do banco de dados antes da inicialização e não iniciarão se a versão estiver fora do intervalo suportado estiver fora do intervalo suportado. Para obter mais detalhes, consulte [databases](#).

Suporte ao PCRE2

O Zabbix agora suporta PCRE e PCRE2. Os pacotes Zabbix para RHEL 7 e mais recentes, SLES (todas as versões), Debian 9 e mais recentes, Ubuntu 16.04 e mais recentes foram atualizados para compilar com PCRE2 em vez de PCRE. Ao compilar a partir de fontes, os usuários podem optar por especificar o sinalizador "--with-libpcre" ou "--with-libpcre2". Se estiver atualizando uma instalação existente, a alteração do PCRE para PCRE2 pode fazer com que algumas expressões regulares se comportem de forma diferente - consulte [Problemas conhecidos](#) para obter detalhes.

**Arquivos de configuração separados

Cada plug-in do Zabbix Agent 2 agora tem um **arquivo de configuração** separado (/manual/appendix/config/zabbix_agent2_plugins). Por padrão, esses arquivos estão localizados no diretório ./zabbix_agent2.d/plugins.d/. O caminho é especificado no parâmetro Include do arquivo de configuração do agente 2 e pode ser relativo ao diretório **zabbix_agent2.conf** ou **zabbix_agent2.win.conf**.

Monitoramento da linha de base

O conjunto de opções de monitoramento de linha de base disponíveis foi ampliado com as duas novas funções **baselinedev** e **baselinewma**.

- **Baselinedev** - compara o último período de dados com os mesmos períodos de dados nas estações anteriores e retorna o número de desvios;
- **baselinewma** - calcula a linha de base calculando a média dos dados do mesmo período de tempo em vários períodos de tempo iguais ("temporadas") usando o algoritmo de média móvel ponderada.

No contexto dessas funções, o termo "estação" refere-se a um período de tempo configurável, que pode ser de horas, dias, semanas, meses ou anos, meses ou anos. A duração de uma temporada e o número de temporadas a serem analisadas são definidos nos parâmetros da função.

Consulte [history functions](#) para obter mais informações.

6 O que há de novo no Zabbix 6.0.1

Aggregate functions The **count_foreach** function now returns '0' for a matching item in the array, if no data are present for the item or the data do not match the filter. Previously such items would be ignored (no data added to the aggregation).

TimescaleDB 2.11 support Support for TimescaleDB version 2.11 is now available.

Configurable TLS and connection parameters in MQTT plugin The **MQTT plugin** for Zabbix agent 2 now provides additional configuration options, which can be defined in the plugin configuration file as **named session** or **default** parameters:

- Connection-related parameters: broker URL, topic, username, and password;
- TLS encryption parameters: location of the top-level CA(s) certificate, MQTT certificate or certificate chain, private key.

All of the new parameters are optional.

JavaScript preprocessing The heap limit for scripts has been upped from 64 to 512 megabytes.

Supported platforms Support for Debian 12 (Bookworm) has been added, and official packages are available for download on [Zabbix website](#).

6 O que há de novo no Zabbix 6.0.1

Agente Zabbix 2 itens

- Suporte nativo para os **itens** **net.dns** e **net.dns.record** foi adicionado. Esses itens, usados com o agente Zabbix 2, agora suportam o processamento de cheques simultâneos. No Windows, endereços IP DNS personalizados são permitidos no parâmetro **ip**, os parâmetros **timeout** e **count** não são mais ignorados.
- **smart.disk.discovery** e **smart.attribute.discovery** **itens**/manual/config/items/itemtypes/zabbix_agent/zabbix_agent2#s.m.a.r.t.), compatível com S.M.A.R.T. plugin, foram atualizados e agora retornam o valor da macro **{#DISKTYPE}** em letras minúsculas.

Discovery of disabled systemd units It is now also possible to discover **disabled** systemd units using the **systemd.unit.discovery** item key, supported by Zabbix agent 2. Note that to have items and triggers created from prototypes for disabled systemd units, it may be necessary to adjust (or remove) prohibiting LLD filters for the **{#UNIT.ACTIVESTATE}** and **{#UNIT.UNITFILESTATE}** macros.

For more details, see [Discovery of systemd services](#).

SNI support in encrypted connections

Encrypted TCP connections between Zabbix agent and Zabbix server or proxy now support SNI.

SourceIP support in LDAP simple checks SourceIP support has been added to LDAP **simple checks**. Note that with OpenLDAP, version 2.6.1 or above is required.

7 O que há de novo no Zabbix 6.0.2

Configuração de Verificação Ativa no Zabbix agent 2

Uma nova opção de configuração **configuration parameter** **ForceActiveChecksOnStart** foi adicionada ao Zabbix agent 2.

Definir o parâmetro como **ForceActiveChecksOnStart=1** garantirá que os dados do item para verificações ativas sejam coletados imediatamente após a reinicialização Zabbix agent iniciar, exceto para itens com Agendamento **update interval**. Caso contrário, a primeira coleta de dados ocorrerá em um momento aleatório, que é inferior ao intervalo de atualização do item, para evitar picos no uso de recurso.

É possível também definir essa opção somente para um plugin específico usando **Plugins.<PluginName>.System.ForceActiveChecksOnStart** (por exemplo, **Plugins.Uptime.System.ForceActiveChecksOnStart=1**). Se definido, o parâmetro de nível de plugin substituirá a configuração global.

Monitoramento JMX O template Generic Java JMX agora contém regras de descoberta para descoberta de baixo nível de pools de memória e coletores de lixo.

Navegação por teclado O controle por teclado foi implementado para os ícones de informações no frontend. Assim, agora é possível focar nos ícones de informações, e abrir dicas usando o teclado.

8 O que há de novo no Zabbix 6.0.3

Métricas PostgreSQL

Um novo **item** foi adicionado ao plugin do PostgreSQL para o agente Zabbix 2. A métrica **pgsql.queries** é utilizada para monitorar o tempo de execução das consultas.

Templates

Um novo template OpenWeatherMap by HTTP agora está disponível, permitindo monitorar o OpenWeatherMap via HTTP. Ver [HTTP template operation](#) para configurações.

As mudanças abaixo foram feitas nos templates existentes:

- Nos templates Windows services pelo Zabbix agent, Windows services pelo Zabbix agent active, Windows pelo Zabbix agent, Windows pelo Zabbix agent active {\$SERVICE.NAME.NOT_MATCHES} o valor da macro foi atualizado para filtrar uma lista de serviços.
- O template PostgreSQL by Zabbix agent 2 agora verificará o número de consultas lentas e irá gerar um problema se a quantidade exceder um limite estabelecido.

Você pode obter esses templates:

- Em Configuration → Templates em novas instalações;
- Se você estiver atualizado a partir de versões anteriores, você pode baixar os novos templates pelo Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente do Zabbix. Em seguida, enquanto estiver em Configuration → Templates você pode importá-los manualmente no Zabbix.

9 O que há de novo no Zabbix 6.0.4

Dados de texto para widget Top hosts Agora é possível selecionar itens com qualquer tipo de informação (incluindo Caractere, Texto, e Log) no widget **Top hosts**. Por exemplo, agora é possível usar este widget para exibir as versões dos agentes do Zabbix em execução em cada host.

Suporte para OpenSSL 3.0 A versão OpenSSL 3.0.x agora é suportada. Observe que essa alteração não afeta a criptografia do frontend (que utiliza seu próprio pacote openssl-php) e as conexões criptografadas JMX do Java gateway para os alvos de monitoramento (que utiliza suas próprias bibliotecas criptografadas em Java).

Templates Novos templates estão disponíveis:

- TrueNAS SNMP - monitoramento de armazenamento TrueNAS OS pelo SNMP
- Proxmox VE pelo HTTP - consultar instruções de configuração para templates [HTTP templates](#)

Novas macros que permitem definir limites de advertência e críticos da utilização do sistema de arquivo para monitoramento de sistemas de arquivos virtuais foram adicionados aos templates

HOST-RESOURCES-MIB armazenamento SNMP, Linux pelo Prom, Linux sistemas de arquivos SNMP, Linux sistemas de arquivos pelo Zabbix agent active, Linux sistema de arquivos pelo Zabbix agent, Mellanox SNMP, PFSense SNMP, Windows sistema de arquivos pelo Zabbix agent active, Windows sistema de arquivos Zabbix agent.

Utilização de arquivos de sistema triggers foram atualizados para utilizar essas macros.

Você consegue obter esses templates:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando de versões anteriores, você pode baixar os novos templates a partir do repositório do Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente do Zabbix. Em seguida, enquanto em Configuração → Templates você pode importá-los manualmente no Zabbix.

Integração GLPI Uma nova integração GLPI [GLPI integration](#) está disponível, permitindo o uso de mídia [webhook](#) para criar problemas na seção de Assistência do GLPI com base em notificações de problemas do Zabbix.

Monitoramento S.M.A.R.T. O Plugin Smart, suportado pelo Zabbix agent 2, agora fornece uma descoberta de disco mais eficiente e permite a obtenção de informações sobre um disco específico, em vez de todos os discos descobertos. Os itens **smart.disk.discovery** e **smart.disk.get** foram atualizados. Os templates SMART pelo Zabbix agent 2 e SMART pelo Zabbix agent 2 (active) também foram modificados para incorporar a nova funcionalidade.

10 O que há de novo no Zabbix 6.0.5

Templates New templates are available:

- CockroachDB by HTTP
- Envoy Proxy by HTTP

See setup instructions for [HTTP templates](#).

You can get these templates:

- In Configuration → Templates in new installations;
- If you are upgrading from previous versions, you can download new templates from Zabbix [Git repository](#) or find them in the `templates` directory of the downloaded latest Zabbix version. Then, while in Configuration → Templates you can import them manually into Zabbix.

Handling of NaN values in Prometheus preprocessing There is a new behavior for handling (skipping) NaN values. So, if a dataset consists of valid numeric values and NaNs, then NaN values are skipped and:

- 'avg', 'max', 'min', 'sum' return a result that is calculated from the valid values
- 'count' returns the number of valid values

If all values in a dataset are NaNs then 'avg', 'max', 'min', and 'sum' return a "no data (at least one value is required)" error, while 'count' returns 0.

Previously, if NaN was the first value in a dataset then:

- 'avg', 'max', 'min', 'sum' returned a "Value "NaN" of type "string" is not suitable for value type "Numeric (float)" error
- 'count' returned the number of values (including NaN values)

Also previously, if NaN was not the first value in a dataset then:

- 'avg', 'sum' returned a "Value "NaN" of type "string" is not suitable for value type "Numeric (float)" error
- 'max' returned the maximum of values until the first NaN was encountered
- 'min' returned the minimum of values until the first NaN was encountered
- 'count' returned the number of values (including NaN values)

Latest data link for hosts shows numbers The latest data link for hosts in [Monitoring -> Hosts](#) now shows the number of items with latest data.

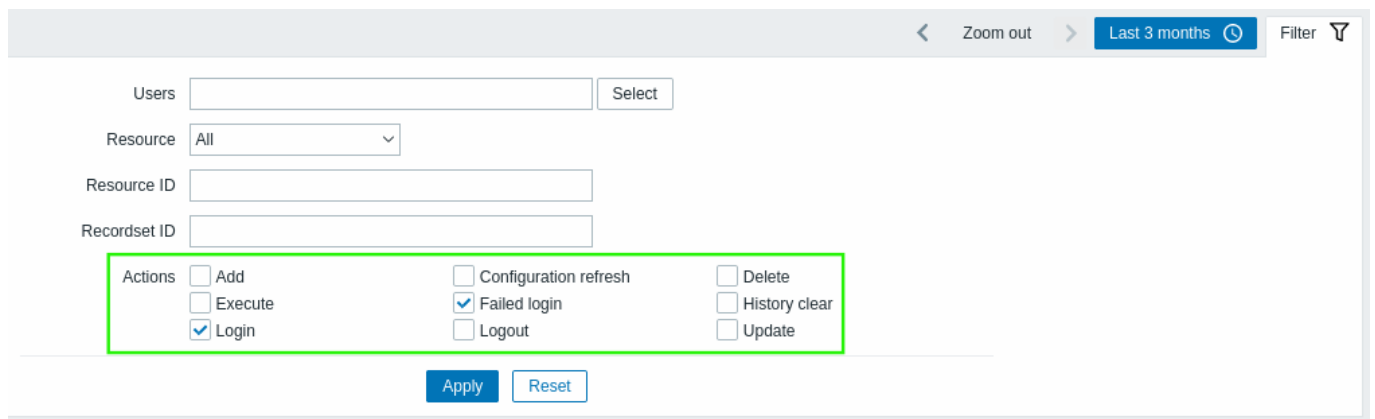
Frontend languages German and Vietnamese languages are now enabled in the frontend.

Expandable lists in latest data subfilter Expandable lists have been introduced in the [latest data](#) subfilter:

- For each entity group (e.g. tags, hosts) 10 rows of entities are now displayed. If there are more entities, a three-dot icon is displayed at the end; if you click on it the list will be expanded to a maximum of 1000 entries (the value of `SUBFILTER_VALUES_PER_GROUP` in [frontend definitions](#)). Previously a non-expandable maximum of 100 entries was the limit.
- In the list of Tag values 10 rows of tag names are now displayed. If there are more tag names with values, a three-dot icon is displayed at the bottom; if you click on it the list will be expanded to a maximum of 200 tag names. Previously, a non-expandable maximum of 20 rows with tag names was the limit.

For each tag name 10 rows of values are displayed (expandable to 1000 entries (the value of `SUBFILTER_VALUES_PER_GROUP` in [frontend definitions](#)))

Audit log filter Multiple actions now can be selected in the audit log filter in Reports -> **Audit**:



The screenshot shows the 'Audit log filter' interface. It has a header with 'Zoom out', 'Last 3 months', and a 'Filter' icon. Below the header are input fields for 'Users', 'Resource' (a dropdown menu), 'Resource ID', and 'Recordset ID'. A 'Select' button is next to the 'Users' field. Below these fields is a section for 'Actions' with a grid of checkboxes: 'Add', 'Execute', 'Login' (checked), 'Configuration refresh', 'Failed login' (checked), 'Logout', 'Delete', 'History clear', and 'Update'. At the bottom of the form are 'Apply' and 'Reset' buttons.

This is useful to see all related actions (for example, successful and failed logins into the frontend) in the audit list.

11 O que há de novo no Zabbix 6.0.6

Suporte PHP 8 Agora, PHP 8.0 e 8.1 são suportados.

Suporte MariaDB 10.7 Agora, a maior versão suportada para Maria DB é 10.7.X.

Plugin carregável MongoDB O MongoDB [plugin](#) não faz mais parte do Zabbix agent 2 e agora está disponível como um plugin carregável. A lista de versões suportadas do MongoDB foram estendidas para 2.6-5.3.

A funcionalidade e o conjunto de itens **items** suportados não foram alterados.

Templates Novos templates

Novos templates estão disponíveis:

- HPE MSA 2040 armazenado por HTTP
- HPE MSA 2060 armazenado por HTTP
- HPE Primera por HTTP

Veja as instruções de configuração para **HTTP templates**.

Você pode acessar esses templates:

- Em Configuration → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você consegue baixar os novos templates do Zabbix em [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente baixada do Zabbix. Em seguida, enquanto em Configuration → Templates, você pode importá-los manualmente no Zabbix.

Alteração no ExpressMS messenger webhook API A versão API foi alterada para v4 no ExpressMS messenger webhook.

12 O que há de novo no Zabbix 6.0.7

Suporte MariaDB 10.8 Agora, a maior versão suportada para Maria DB é 10.8.X.

Suporte TimescaleDB 2.6 Agora, a maior versão suportada para TimescaleDB é 2.6.

Templates Novos templates

Um novo template HPE Synergy por HTTP está disponível.

Veja as instruções da configuração para **HTTP templates**.

Você consegue acessar esse template:

- Em Configuration → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você pode baixar os novos templates de Zabbix em [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente baixada do Zabbix. Em seguida, enquanto em Configuration → Templates, você pode importá-los manualmente no Zabbix.

Templates atualizados

O template PostgreSQL Agent 2 foi atualizado. Uma trigger para detectar falhas no checksum foi adicionada ao item Dbstat do [PostgreSQL Agent 2 template](#).

Você consegue acessar esse template:

- Em Configuration → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você consegue baixar os novos templates do repositório do Zabbix [Git repository](#) ou encontrá-los templates no diretório da versão mais recente baixada do Zabbix. Em seguida, enquanto em Configuration → Templates, você pode importá-los manualmente no Zabbix.

13 O que há de novo no Zabbix 6.0.8

Mês abreviado com letra maiúscula Agora, um "mês" é abreviado com letra maiúscula "M" no frontend. Anteriormente, era abreviado com letra minúscula "m", sobrepondo a abreviação do minuto.

Suporte TimescaleDB 2.7 Agora, a versão maior suportada para TimescaleDB é 2.7.

Templates Um novo template OPNsense por SNMP [template](#) está disponível.

Você pode acessar esse template:

- Em Configuration → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você pode pode baixar novos templates do Zabbix em [Git repository](https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates) (<https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates>) ou encontrá-los no diretório da versão mais recente baixada do Zabbix. Em seguida, enquanto em Configuration → Templates, você pode importá-los manualmente no Zabbix.

Pacotes RHEL renomeados Os pacotes RHEL foram renomeados, adicionando uma palavra "release" no nome:

Naming	Nome do pacote
Old	zabbix-agent-6.0.7-1.el9.x86_64.rpm
New	zabbix-agent-6.0.8-release1.el9.x86_64.rpm

Não há nenhuma mudança funcional associada a essa alteração. Isso é necessário como preparação para fornecer pacotes da versão menor (ou seja, 6.0.x) de candidatos a lançamento, esperados para começar com o 6.0.9. A mudança do nome garantirá que, para alguém que tenha tanto os repositórios estáveis e instáveis habilitados no seu sistema, as atualizações de repositório sejam recebidas na ordem correta. Essa mudança no nome é somente para pacotes RHEL.

14 O que há de novo no Zabbix 6.0.9

Macros de expressão Agora, os macros {ITEM.KEY<1-9>} são suportados dentro dos [expression macros](#).

Pacotes Os scripts SQL foram movidos do diretório /usr/share/doc para /usr/share nos pacotes Zabbix.

15 O que há de novo no Zabbix 6.0.10

Configurações de filtro memorizadas

Em várias páginas de Monitoramento (Problemas, Hosts, Últimos dados), as configurações de filtro atuais agora são lembradas no perfil do usuário. Quando o usuário abre a página novamente, as configurações do filtro continuarão da mesma forma. Além

disso, a marcação de um filtro favorito alterado (mas não salvo) agora é um ponto ao lado no nome do filtro, em vez do nome do filtro em não itálico.

Suporte TimescaleDB 2.8 A maior versão suportada para TimescaleDB agora é de 2.8.

Suporte PostgreSQL 15 Agora o PostgreSQL 15 é suportado. Perceba que o TimescaleDB ainda **não** suporta PostgreSQL 15.

Possibilidade de construir Zabbix agent 2 offline O Zabbix agent 2 agora pode ser construído offline. O tarball de origem agora inclui o diretório `src/go/vendor`, que deve verificar que a golang não é obrigada a baixar os módulos de dependência automaticamente. É possível atualizar os últimos módulos manualmente, usando os comandos `go mod tidy` ou `go get`.

Plugin PostgreSQL carregável O plugin PostgreSQL **plugin** agora é carregável no Zabbix agent 2 (anteriormente integrado).

Veja também: repositório do [PostgreSQL loadable plugin](#)

Frontend Diversos

- Alertas sobre configuração incorreta de manutenção para o TimescaleDB agora são exibidos se as tabelas de histórico ou trend tiverem blocos comprimidos, mas as opções `Override item history period` ou `Override item trend period` estiverem desabilitadas. Para mais informações, veja [TimescaleDB setup](#).

16 O que há de novo no Zabbix 6.0.11

Reportando sistemas de arquivo com inodes igual a zero Os itens do agente `vfs.fs.get` agora são capazes de reportar sistemas de arquivo com contagem de inodes igual a zero, o que pode ser o caso para sistemas de arquivo com inodes dinâmicos. (por exemplo: btrfs).

Ainda, agora os itens `vfs.fs.inode` não se tornarão não suportados em casos como, modo configurado `'pfree'` ou `'pused'`. Em vez disso, os valores de `pfree/pused` para tais arquivos serão reportados como `"100"` e `"0"`, respectivamente.

Consultas utilizadas da API Consultas de banco de dados da API, usadas ao pesquisar por nomes nas tabelas `hosts` e `items`, foram otimizadas e agora serão processadas de forma mais eficiente.

17 O que há de novo no Zabbix 6.0.12

Melhoria de desempenho nos sincronizadores de histórico O desempenho de sincronizadores de histórico foi aprimorado pela introdução de uma nova trava de leitura e escrita. Isso reduz o travamento entre os sincronizadores de histórico, `trappers` e `proxy poller`, usando uma trava de leitura compartilhada ao acessar o cache de configuração. A nova trava pode ser bloqueada somente para escrita pelo sincronizador de configuração que realiza a recarga do cache de configuração.

18 O que há de novo na Zabbix 6.0.13

Alterações (breaking changes) Versão de plugins carregáveis

Loadable plugins para o Zabbix agente 2 agora é utilizado o mesmo sistema de versionamento do próprio Zabbix. As seguintes alterações foram feitas:

- MongoDB 1.2.0 -> MongoDB 6.0.13
- PostgreSQL 1.2.1 -> PostgreSQL 6.0.13

Esses plugins são compatíveis com qualquer versão secundária do Zabbix 6.0.

Observe que o repositório de fonte para cada plugin agora contém um branch dedicado `release/6.0` branch (anteriormente, havia apenas o branch `master`).

Suporte MariaDB 10.10 A versão máxima suportada para MariaDB agora é 10.10.X.

Importação de configuração Anteriormente, o processo de importação falharia devido a uma incompatibilidade de UUID de uma entidade importável (grupo host, item, graph, etc.). Por exemplo, não era possível importar o grupo host se algum grupo host já tivesse um mesmo nome existente no host.

Na nova versão, a importação falhará por causa de uma incompatibilidade de UUID; em vez disso, a entidade será correspondida pelos critérios de singularidade, como o ID da entidade. A entidade será importada, e o UUID será atualizado para o UUID da entidade importada.

Outra melhoria, ao remover o vínculo do template (a opção Excluir ausentes para o vínculo do template) através do template importado ou host, as entidades herdadas não são mais removidas (o template fica desvinculado, não desvinculado e limpo), a menos que essas entidades estejam ausentes no arquivo de importação entidades e a opção Excluir ausentes para a entidade específica esteja marcada.

Devido a essa mudança, a opção de mensagem de alerta ao marcar a opção Excluir ausentes para o vínculo do template, não será mais exibida.

Consultar tablespaces separadas em bancos de dados Oracle com Zabbix agent 2 O seguinte [Zabbix agent 2 items](#), suportados para o plugin do Oracle, agora possuem parâmetros opcionais adicionais:

- `oracle.diskgroups.stats[<existingParameters>,<diskgroup>]`
- `oracle.archive.info[<existingParameters>,<destination>]`
- `oracle.cdb.info[<existingParameters>,<database>]`
- `oracle.pdb.info[<existingParameters>,<database>]`
- `oracle.ts.stats[<existingParameters>,<tablespace>,<type>]`

Esses parâmetros permitem consultar instanciadas separadas de dados específicos, ao invés de todos os dados, melhorando assim o desempenho.

Recuperar informação adicional com `docker.container_info[]` O `docker.container_info[]` [Zabbix agent 2 item](#) agora suporta a opção de recuperar informações de nível baixo de forma parcial (short) ou recuperar informações de forma completa sobre um Docker container.

Comandos de tempo de execução do para profiling Os comandos de tempo de execução para profiling foram adicionados ao Zabbix server e ao Zabbix proxy.

- `prof_enable` - habilitar profiling
- `prof_disable` - desabilitar profiling

O Profiling por ser habilitado pelo processo server/proxy. Profiling habilitados fornecem detalhes de todo rwlocks/mutexes pela função nome.

Veja também:

- [Zabbix server runtime commands](#)
- [Zabbix proxy runtime commands](#)

Função HMAC para JavaScript Uma nova função foi adicionada ao mecanismo JavaScript permitindo retornar um hash HMAC:

- `hmac('<hash type>',key,string)`

Essa função é útil em casos em que é necessário um código de autenticação de mensagem baseado em hash (HMAC) para assinar solicitações. Os tipos de hash MD5 e SHA256 são suportados, e. g.:

- `hmac('md5',key,string)`
- `hmac('sha256',key,string)`

Templates Novos templates estão disponíveis:

- AWS EC2 via HTTP
- AWS via HTTP
- Instância AWS RDS via HTTP
- Bucket AWS S3 via HTTP
- Azure via HTTP
- Servidor Control-M via HTTP
- Gerenciador empresarial Control-M via HTTP
- Veeam Backup Gerenciador empresarial via HTTP
- Veeam Backup e Replication via HTTP

Veja instruções de setup para [HTTP templates](#).

O template [Oracle by Zabbix agent 2](#) foi atualizado (itens estáticos múltiplos removidos; múltiplos itens prototypes adicionados) de acordo com a mudança realizada [Zabbix agent 2 items](#).

Para mais informações sobre atualizações, veja [Template changes](#).

Você tem acesso a esses templates em:

- Em Configuração → Templates em nova instalações;
- se você está atualizando pelas versões anteriores, você pode fazer download dos novos templates pelo Zabbix [Git repository](#) ou encontrar eles no diretório de downloads da última versão da Zabbix. Depois, enquanto estiver na Configuração → Templates, você pode importá-los manualmente no Zabbix.

Suporte TimescaleDB 2.9 A versão máxima suportada para TimescaleDB agora é 2.9.

Integrações do Webhook Um novo tipo de mídia LINE agora está disponível, permitindo o uso da funcionalidade [webhook](#) para enviar notificações sobre eventos do Zabbix para o mensageiro LINE.

Idiomas Frontend Idiomas como Catalão e Romeno agora estão habilitadas no frontend.

Biblioteca Golang para atualização do Windows A biblioteca Golang usada pelo Zabbix agent 2 em conjunto com os plugins MongoDB ou PostgreSQL para monitorar Windows agora é [github.com/Microsoft/go-winio](#), versão 0.6.0 (anteriormente [github.com/natefinch/npipes](#)). Veja também [Golang libraries](#), [MongoDB plugin dependencies](#), e [PostgreSQL plugin dependencies](#).

Aumento do limite de descritores de arquivos abertos para Zabbix agent 2 The system service file shipped in Zabbix agent 2 packages now declares the open file descriptor limit of 8196. Anteriormente, o limite do sistema padrão de 1024 foi utilizado. O novo limite é suficiente para a configuração padrão Zabbix agent 2. Se você tem uma configuração não padrão do agent 2 configuration, por exemplo use plugins adicionais ou recursos estendidos, esse limite pode precisar ser aumentado manualmente futuramente. Nesse caso, ajuste o parâmetro `LimitNOFILE` no arquivo de unidade do sistema.

19 O que há de novo no Zabbix 6.0.14

Plugins carregáveis Conexão plugin Encrypted MongoDB

O plugin MongoDB agora oferece suporte à criptografia TLS ao se conectar ao MongoDB utilizando sessões nomeadas. O plugin atualizado (Plugin MongoDB 1.2.1) está incluído nos pacotes oficiais do Zabbix a partir da versão Zabbix 6.0.14. Observe que o MongoDB é um plugin carregável e pode ser instalado separadamente a partir dos pacotes ou de fontes. O plugin funcionará com qualquer versão inferior do Zabbix 6.0. Para mais detalhes, ver [MongoDB plugin](#).

Suporte PHP Agora, a maior versão suportada para PHP é 8.2.

Limites para objetos JavaScript no pré-processamento Os seguintes limites para [JavaScript objects](#) no pré-processamento foram inseridos:

- O tamanho total de todas as mensagens que pode ser carregado com o método `log()` foi limitado para 8MB por execução de script.
- A inicialização de múltiplos objetos `HttpRequest` foi limitada a 10 por execução de script.
- O tamanho total do comprimento dos campos de cabeçalho que podem ser adicionados a um único objeto `HttpRequest` com o método `addHeader()` foi limitado a 128 Kbytes (caracteres especiais e nomes de cabeçalho estão incluídos).

20 O que há de novo no Zabbix 6.0.15

MariaDB 10.11 support A versão máxima [supported version](#) para MariaDB agora é 10.11.X.

Suporte TimescaleDB 2.10 O máximo [supported version](#) para TimescaleDB agora é 2.10.

Opções de conexão para Oracle plugin O Oracle plugin, suportado para Zabbix agent 2, agora permite especificar a opção de login as `sysdba`, as `sysoper`, ou as `sysasm`. A opção pode ser adicionada tanto parâmetro chave de item do usuário quanto ao parâmetro de configuração do plugin. `Plugins.Oracle.Sessions.<SessionName>`. Usuário do formato `user as sysdba` (a opção de login é case-insensitive; não deve conter um espaço final).

Assinando dados usando RS256 A nova função do JavaScript `sign(hash,key,data)` foi implementada permitindo o uso do algoritmo de criptografia RS256 para calcular a assinatura. Para mais detalhes ver: [Additional JavaScript objects](#).

21 O que há de novo no Zabbix 6.0.16

Otimização de sincronização de configuração para Oracle

Para instalação do Zabbix com Oracle, agora é possível alterar manualmente os tipos de campo de item e pré-processamento de item de `nclob` para `nvarchar2` aplicando um patch de banco de dados.

22 O que há de novo no Zabbix 6.0.17

Integrações Webhook Um novo tipo de mídia Webhook `webhook` foi adicionado para enviar notificações do Zabbix para o [Event-Driven Ansible](#).

Misturar parâmetros da chave do item e da sessão em plugins do Zabbix agent 2 Agora, o Zabbix agent 2 permite a substituição dos parâmetros `named session`, especificando os novos valores nos parâmetros da chave do item. Anteriormente, os usuários precisavam selecionar se preferiam fornecer os valores da string de conexão em uma sessão nomeada ou em uma chave do item. Se as sessões nomeadas fossem usadas, os parâmetros relacionados à chave do item precisavam estar vazios. Agora, quando usamos sessões nomeadas, somente o primeiro parâmetro (geralmente, uma URI) deve ser especificado na sessão nomeada, enquanto outros parâmetros podem ser definidos na sessão nomeada ou na chave do item.

O suporte a HTML na atribuição do Geomap foi removido Agora, o texto de atribuição para o `Geomap dashboard widget` pode somente conter um texto simples; o suporte para HTML foi removido.

Nas configurações `Geographical maps` da Administração → Sessão geral, o campo Atribuição somente estará visível quando o Tile provider estiver configurado como Outro.

23 O que há de novo no Zabbix 6.0.18

Items `docker.container_stats`

O item `docker.container_stats` no Zabbix agent 2 agora também retorna uma propriedade `pids_stats`, que fornece o número atual de processos/threads no container.

Exportação de configuração mais limpa Os arquivos YAML gerados durante a exportação de configurações de entidades no Zabbix não contêm mais linhas vazias entre as entidades em um array, o que torna esses arquivos menores e mais convenientes de trabalhar. Consulte a seção [Configuration export/import](#) para exemplos atualizados de exportação.

UTF-8 BOM na importação de configuração A importação da configuração `Configuration import` agora suporta arquivos com uma marca de ordem UTF-8 byte- (BOM).

Monitoramento Cosmos DB O template Azure by HTTP agora também funciona com Azure Cosmos DB para MongoDB.

Você acessa esse template em:

- Em Configuração → Templates em novas instalações
- Se você estiver atualizando a partir de versões anteriores, você pode baixar esse template do repositório do Zabbix [Git repository](#) ou encontrá-lo no diretório de `zabbix/templates` da versão mais recente do Zabbix baixada. Em seguida, enquanto estiver em Configuração → Templates, você pode importá-lo manualmente no Zabbix.

Manutenção de histórico de proxy A limitação na quantidade de informações desatualizadas excluídas do banco de dados do proxy por ciclo de manutenção do histórico de proxy foi removida. Anteriormente, o `housekeeper` excluía não mais que 4 vezes as horas `HousekeepingFrequency` de informações desatualizadas. Por exemplo, se `HousekeepingFrequency` estivesse configurado como "1", não era excluído mais do que 4 horas de informações desatualizadas (iniciando a partir da entrada mais antiga). Nos casos em que um proxy constantemente recebesse dados mais antigos do que definido em `ProxyOfflineBuffer`, isso poderia resultar em um acúmulo excessivo de dados. Agora essa limitação foi removida, proporcionando uma solução de gerenciamento de proxy mais eficaz.

Monitoramento da plataforma Google Cloud Um novo [template](#) para Google Cloud Platform por HTTP (GCP por HTTP) está disponível.

Veja instruções de configuração para [HTTP templates](#).

Você pode acessar esse template:

- Em Configuração → Templates em novas instalações.
- Se você estiver atualizando de versões anteriores, você pode baixar esse template do repositório de Zabbix [Git repository](#) ou encontrá-lo no diretório `zabbix/templates` da versão mais recente do Zabbix. Em seguida, enquanto em Configuração → Templates, você pode importá-lo manualmente no Zabbix.

Valores padrão para Zabbix agent 2 Os plugins Zabbix agent 2 agora permitem definir os valores padrão para a conexão com tags de monitoramento no arquivo de configuração. Se nenhum valor for definido em um item key ou em uma sessão nomeada, o plugin utilizará o valor definido no parâmetro padrão correspondente.

Os novos parâmetros possuem a estrutura `Plugins.<PluginName>.Default.<Parameter>` - por exemplo, `Plugins.MongoDB.Default`. Para ter mais informações, acesse:

- [Configuring plugins](#)
- [Plugin configuration file parameters](#)

25 O que há de novo no Zabbix 6.0.20

Templates Novos templates estão disponíveis:

- [AWS ECS Cluster by HTTP](#) (acompanhado com seu [Serverless Cluster version](#))
- [Cisco SD-WAN by HTTP](#)
- [OpenStack by HTTP](#), que inclui template OpenStack Nova por HTTP para monitoramento de OpenStack Nova service
- [PostgreSQL by ODBC](#)

Você pode acessar esses templates:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando de versões anteriores, você pode baixar novos templates do repositório de Zabbix [Git repository](#) ou encontrá-los no diretório `zabbix/templates` da versão mais recente de Zabbix baixada. Em seguida, enquanto em Configuração → Templates você pode importá-los manualmente no Zabbix.

Frontend Verificação ortográfica desativada em áreas de texto não descritivas

A verificação ortográfica foi desativada para áreas de textos em que textos não descritivos são digitados, como scripts, expressões, valores macro, etc.

####Diversos

Conexão TLS com banco de dados para MySQL no SLES 12

Os pacotes para instalação do servidor/proxy no SUSE Linux Enterprise Server versão 12 agora são construídos usando a biblioteca MariaDB Connector/C, possibilitando assim a criptografia da conexão com o MySQL utilizando o parâmetro `DBTLSConnect` [parameter](#). Os valores de criptografia suportados são "required" e "verify_full".

26 O que há de novo no Zabbix 6.0.21

Suporte para MySQL 8.1

Agora, a versão máxima suportada [supported version](#) para MySQL é 8.1.X.

Suporte para MariaDB 11.0

Agora, a versão máxima suportada **supported version** para MariaDB é 11.0.X.surrounding-software) for MariaDB is now 11.0.X.

Log file monitoring

Monitoramento Log file

Para os itens `log[]`, `logrt[]`, `log.count[]`, `logrt.count[]`, erros de tempo de execução de expressões regulares agora são registrados no arquivo de log do agente Zabbix. Para mais detalhes, consulte **more details**.

Itens Item novo para o agente 2 Zabbix

Um novo item foi adicionado ao plugin MySQL para o agente 2 Zabbix. Esse novo item **mysql.custom.query**, pode ser utilizado para executar consultas MySQL personalizadas.

Templates Um novo template está disponível:

- [AWS Cost Explorer by HTTP](#)

Você pode obter esse template:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você pode baixar os novos templates do repositório de Zabbix [Git repository](#) ou encontrá-los no diretório de downloads zabbix/templates da versão mais recente do Zabbix. Em seguida, enquanto em Configuração → Templates você pode importá-los manualmente no Zabbix.

Notificações Integrações Webhook

Foi adicionado um novo tipo de mídia **webhook** para enviar notificações do Zabbix para o [Mantis Bug Tracker](#).

Instalações Suporte para ARM64/AArch64

Agora, os pacotes de instalação ARM64/AArch64 estão disponíveis para Debian, RHEL 8, 9 seus derivados, como também para SLES/OpenSUSE Leap 15.

27 O que há de novo no Zabbix 6.0.22

Funções agregadas

Agora, a função **last_foreach** também é suportada nas seguintes funções agregadas **aggregate functions**: **kurtosis**, **mad**, **skewness**, **stddevpop**, **stddevsamp**, **sumofsquares**, **varpop**, e **varsamp**.

Limite do valor de retorno

O limite do valor de retorno para receber o dado de fontes externas (como de scripts ou outros programas) foi aumentado para 16MB. Isso afeta:

- Itens Agent **system.run[]** e **vfs.file.contents[]**
- Verificações de agentes personalizados definidos em **user parameters**
- **SSH agent**, **External check**, e itens **Script**
- **Remote commands**

Templates Novos templates estão disponíveis:

- [Acronis Cyber Protect Cloud by HTTP](#)
- [HashiCorp Nomad by HTTP](#)
- [MantisBT by HTTP](#)

Você pode obter esses templates:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você pode baixar os novos templates de Zabbix [Git repository](#) ou encontrá-los no diretório de últimos acessos zabbix/templates. Em seguida, enquanto em Configuration → Templates você pode importá-los manualmente para o Zabbix.

28 O que há de novo no Zabbix 6.0.23

Consulte [breaking changes](#) para essa versão.

Bancos de dados Versões suportadas

Agora PostgreSQL **16** e MariaDB **11.1** são suportados. Veja também [Requirements](#).

Plugins Novo item PostgreSQL Zabbix agent 2 plugin

Um novo item [item](#), **pgsql.version**, foi adicionado ao plugin Zabbix Agent 2 para PostgreSQL. Esse item é usado para retornar a versão do PostgreSQL.

Templates Novos templates

Novos templates estão disponíveis:

- [FortiGate by HTTP](#)
- [FortiGate by SNMP](#)
- [Nextcloud by HTTP](#)

Você consegue obter esses templates:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você consegue baixar os novos templates a partir do diretório do Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente baixada do Zabbix. Em seguida, enquanto em Configuração → Templates, você pode importá-los manualmente do Zabbix.

Templates atualizados

Agora os templates [PostgreSQL by ODBC](#) e [PostgreSQL by Zabbix agent 2](#) incluem o item e o trigger para monitorar a versão PostgreSQL.

[Cisco Meraki organization by HTTP](#) O template foi complementado com itens, itens prototypes, regras de LLD, e macros relacionados a autenticação, licenças, network, SAML roles, e status de VPN.

Frontend Diversos

O botão Limpar histórico localizado em Configuração → Hosts → [Items](#) foi renomeado para Limpar histórico e trends para descrever mais precisamente sua função, que é a mesma do botão Limpar histórico e trends no [configuration form](#).

Na configuração [trigger action](#), a condição Trigger nome foi renomeada Evento nome para descrever melhor sua função. Observe que por padrão, o evento nome é combinado com o trigger nome, a menos que um evento nome personalizado seja especificado em [trigger configuration](#).

29 O que há de novo no Zabbix 6.0.24

Banco de dados Suporte para TimescaleDB 2.12

O suporte para TimescaleDB versão 2.12 agora está disponível.

Plugins Novos itens nos plugins do Zabbix agent 2

Os itens para retornar a versão do servidor do banco de dados agora estão disponíveis em [MongoDB plugin](#) ([mongodb.version](#)) e em [Oracle Database plugin](#) ([oracle.version](#)).

Itens Conversão de conteúdo para UTF-8

Itens do agente HTTP, cenários web, verificações web e itens JavaScript foram melhorados para converter para UTF-8 a partir do conjunto de caracteres especificado no cabeçalho HTTP or na meta tag HTTP.

Templates Novos templates

Um novo template está disponível:

- [HPE iLO by HTTP](#)

Você pode obter esse template:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando de versões anteriores, você pode baixar os novos templates a partir do diretório do Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates da versão mais recente do Zabbix. Em seguida, enquanto em Configuração → Templates, você pode importá-los manualmente no Zabbix.

Templates atualizados

Integração com OpenShift foi adicionado ao template [Kubernetes cluster state by HTTP](#).

30 O que há de novo em Zabbix 6.0.25

Suporte TimescaleDB 2.13

O suporte para TimescaleDB version 2.13 agora está disponível.

Itens Opções adicionais para ssh.run

O item **ssh.run[]** foi atualizado e agora permite passar opções adicionais de SSH como parte da item key. Essas opções são suportadas apenas utilizando biblioteca libssh na versão 0.9.0 ou superior ou biblioteca libssh2. As keys e valores de opções suportados dependem da biblioteca SSH. Para mais detalhes, consulte [SSH checks](#).

Plugins Parâmetro de modo de Cache para plugin PostgreSQL

Novos parâmetros para controlar o modo de cache por padrão ou no nível do nome da sessão foram adicionados à configuração do plugin do PostgreSQL [configuration](#):

- `Plugins.PostgreSQL.Default.CacheMode`
- `Plugins.PostgreSQL.Sessions.<SessionName>.CacheMode`

O parâmetro de modo de cache pode ter um ou dois valores permitidos: preparar (padrão) ou descrever. Observe que "descrever" é principalmente útil quando o ambiente não permite declarações preparadas, como ao executar um pool de conexões como o PgBouncer.

31 O que há de novo no Zabbix 6.0.26

Templates Templates novos

O conjunto de templates [Azure by HTTP](#) foi complementado com o Azure Cost Management pelo HTTP template.

Você pode acessar esse template:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando a partir de versões anteriores, você pode baixar os templates novos a partir do repositório do Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates de downloads da versão mais recente do Zabbix. Em seguida, enquanto em Configuration → Templates, você pode importá-los manualmente no Zabbix.

Templates atualizados

Template [MSSQL by ODBC](#) :

- um novo item foi adicionado - MSSQL DB '#{DBNAME}': Modelo de recuperação, que retorna o modelo de recuperação do banco de dados durante a descoberta do banco de dados;
- macros novos, nomeados `{MSSQL.BACKUP_FULL.USED}`, `{MSSQL.BACKUP_DIFF.USED}`, `{MSSQL.BACKUP_LOG.USED}`, foram adicionados - esses podem ser usados para desativar os triggers de idade do backup para um determinado banco de dados;
- regras de LLD para quórum e descoberta de membros do quórum foram adicionados;
- o tipo das regras de LLD mudaram de "Database monitor" para "Dependent item";
- itens com chave `db.odbc.discovery` tornaram-se dependente do item `db.odbc.get`.

Frontend PHP support

The maximum supported version for PHP is now 8.3.

X-Frame-Options HTTP header

The X-Frame-Options header parameter has been renamed to Use X-Frame-Options header, now consists of a checkbox and an input field (allowing you to disable the header by unmarking a checkbox instead of specifying "null" in the input field), and supports additional values.

Other security parameters now also follow the same structure. For more information, see the [security](#) parameters in Administration → General.

Databases MySQL 8.2 support

The maximum [supported version](#) for MySQL is now 8.2.X.

32 O que há de novo no Zabbix 6.0.27

Databases MySQL 8.3 support

The maximum [supported version](#) for MySQL is now 8.3.X.

Suporte MariaDB 11.2

A versão máxima suportada [supported version](#) para MariaDB agora é 11.2.X.

Plugins MSSQL

A new plugin for direct monitoring of MSSQL by Zabbix agent 2 has been added.

For more information, see:

- [MSSQL plugin readme](#)
- [Agent 2 items](#)
- [MSSQL plugin parameters](#)

Templates Templates novos

Um template novo está disponível:

- [YugabyteDB by HTTP](#), que inclui o template YugabyteDB Cluster por HTTP para monitorar cada cluster YugabyteDB.

Você pode obter esse template em:

- Em Configuração → Templates em novas instalações;
- Se estiver atualizando a partir de versões anteriores, você pode baixar os templates novos a partir do repositório de templates do Zabbix [Git repository](#) ou encontrá-los no diretório de zabbix/templates da última versão do Zabbix baixada. Então, enquanto estiver em Configuração → Templates, você pode importá-los manualmente para o Zabbix.

Platforms Zabbix agent 2 support on Windows

To prevent critical security vulnerabilities, the minimum Windows version for Zabbix agent 2 has been raised to Windows 10/Windows Server 2016. See note under [Supported platforms](#) for more information.

33 O que há de novo no Zabbix 6.0.28

Banco de dados Suporte ao TimescaleDB 2.14

A versão máxima suportada [supported version](#) para o TimescaleDB agora é 2.14.X.

Templates Novos templates

Novos templates estão disponíveis em:

- [AWS ELB Application Load Balancer by HTTP](#)
- [Check Point Next Generation Firewall by SNMP](#)
- [MSSQL by Zabbix agent 2](#)

Você pode obter esses templates:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando de versões anteriores, pode baixar os novos templates a partir do repositório Git do Zabbix [Git repository](#) ou encontrá-los no diretório `zabbix/templates` da versão mais recente do Zabbix que você baixou. Em seguida, enquanto estiver em Configuração → Templates, você pode importá-los manualmente para o Zabbix.

34 O que há de novo no Zabbix 6.0.29

Banco de dados Suporte para MariaDB 11.3

A versão máxima suportada **supported version** para MariaDB agora é 11.3.X

Templates Novos templates

Um novo template está disponível:

- [Oracle Cloud by HTTP](#), um template principal que descobre vários serviços e recursos da Infraestrutura de Nuvem da Oracle (OCI).

Você pode obter este template:

- Em Configuração → Templates em novas instalações;
- Se você estiver atualizando de versões anteriores, pode baixar novos templates do repositório do Zabbix [Git repository](#) ou encontrá-los no diretório `zabbix/templates` da última versão do Zabbix que você baixou. Em seguida, enquanto estiver em Configuração → Templates, , você pode importá-los manualmente para o Zabbix.

Templates atualizados

- O template [FortiGate by SNMP](#) foi complementado com métricas relacionadas a VPN, alta disponibilidade (HA), pontos de término sem fio (WTPs), verificações de saúde SD-WAN e sensores de hardware.
- O template [MySQL by ODBC](#) foi complementado com os itens "MySQL: Obter banco de dados" e "MySQL: Obter replicação". As regras LLD "Descoberta de banco de dados" e "Descoberta de replicação" foram alteradas para o tipo "Item dependente".
- O template [Oracle by ODBC](#) foi complementado com os itens "Oracle: Obter log de arquivo", "Oracle: Obter grupos de disco ASM", "Oracle: Obter banco de dados", foi complementado com os itens "Oracle: Obter log de arquivo", "Oracle: Obter grupos de disco ASM", "Oracle: Obter banco de dados",
- O template The VMware Hypervisor dentro dos conjuntos de templates [VMware](#) e [VMware FQDN](#) foi complementado com uma nova regra LLD, "Descoberta de sensores".

35 O que há de novo no Zabbix 6.0.30

Frontend Idiomas do Frontend

Os idiomas Holandês, Georgiano e Espanhol estão agora habilitados no frontend.

Plugins Ember+

Um novo plugin para monitoramento direto do Ember+ pelo Zabbix Agent 2 foi adicionado.

Para mais informações, consulte::

- [Ember+ plugin readme](#)
- [Agent 2 items](#)
- [Ember+ plugin parameters](#)
- [Agent 2 installation](#)

Templates Novos templates

O conjunto de templates AWS ELB foi complementado com o template [AWS ELB Network Load Balancer by HTTP](#).

Você pode obter este template:

- Em Configuration → Templates em novas instalações;
- Se estiver atualizando de versões anteriores, você pode baixar novos templates do repositório do Zabbix [Git repository](#) ou encontrá-los no diretório zabbix/templates da última versão do Zabbix que você baixou. Em seguida, enquanto estiver em Configuration → Templates, você pode importá-los manualmente para o Zabbix.

Updated templates

The [OS templates](#) (agent, SNMP, and Prometheus-based) have been given a mounted filesystem update. In mounted filesystem discovery, the "Space is low" and "Space is critically low" triggers no longer have the absolute threshold and burst condition.

Macros Macro support for email media types

When configuring an [email media type](#), it is now possible to use macros in the [username and password fields](#).

36 What's new in Zabbix 6.0.31

Frontend Preprocessing test result truncation

When [testing preprocessing steps](#), test results are now truncated to a maximum size of 512KB when sent to the frontend. Note that data larger than 512KB is still processed fully by Zabbix server.

2. Definições

Visual geral Nesta seção você pode aprender o significado de alguns termos comumente usados no Zabbix.

Definições **host**

- um dispositivo ligado à rede que você queira monitorar, com IP/DNS.

grupo de host

- um agrupamento lógico de hosts; ele pode conter hosts e modelos. Hosts e modelos dentro de um grupo de host não são de nenhuma forma associados um ao outro. Grupos de host são usados quando garantindo direitos de acesso aos hosts para diferentes grupos de usuário.

item

- um fragmento de dado em especial que você queira receber de um host, uma métrica de dados.

pré-processamento de valor

- *uma transformação de um valor de métrica recebido antes de salvá-lo no banco de dados.

gatilho

- uma expressão lógica que define um limite (threshold) de problema e é usado para "avaliar" os dados recebidos em itens.

Quando os dados recebidos estão acima do limite, os gatilhos vão do estado 'Ok' para 'Problema'. Quando os dados recebidos estão abaixo do limite, os gatilhos se mantêm/retornam para um estado 'Ok'.

evento

- uma ocorrência única de algo que mereça atenção tal como um gatilho mudando de estado ou um autoregistro de descoberta/agente acontecendo.

etiqueta de evento

- um marcador pré-definido para o evento. Ela pode ser usada na correlação de evento, granulação de permissão, etc.

correlação de evento

- um método de correlação de problemas às suas soluções flexivelmente e precisamente.

Por exemplo, você pode definir que um problema reportado por um gatilho pode ser solucionado por outro gatilho, que pode até mesmo usar método de coleta de dados diferente.

problema

- um gatilho que está em estado de "Problema".

atualização de problema

- opções de gerenciamento de problema fornecidas pelo Zabbix, tais como adição de comentário, reconhecimento, mudança de severidade ou encerrar manualmente.

ação

- meios pré-definidos de reagir a um evento.

Uma ação consiste de operações (p.e. enviando uma notificação) e condições (quando a operação é executada)

escalação

- um cenário customizado para execução de operações dentro de uma ação; uma sequência de envio de notificações/execução de comandos remotos.

mídia

- um meio de entrega de notificações; canal de entrega.

notificação

- uma mensagem sobre algum evento enviada para um usuário através do canal de mídia escolhido.

comando remoto

- um comando pré-definido que é automaticamente executado em um host monitorado diante de alguma condição.

modelo (template)

- um conjunto de entidades (itens, gatilhos, gráficos, regras de descoberta de baixo-nível, cenários web) prontas para serem aplicadas a um ou vários hosts.

A função dos modelos é acelerar a implementação de tarefas de monitoramento em um host; também tornar mais fácil a aplicação de mudanças em massa às tarefas de monitoramento. Modelos são associados diretamente a hosts individualmente.

cenário web

- uma ou várias requisições HTTP para verificar a disponibilidade de um site web.

frontend

- a interface web fornecida com o Zabbix.

dashboard

- seção customizável da interface web exibindo resumos e visualizações de informações importantes em unidades visuais chamadas widgets.

widget

- unidade visual exibindo informações de um certo tipo e origem (um sumário, um mapa, um gráfico, o relógio, etc), usado no dashboard.

Zabbix API

- A API do Zabbix permite usar o protocolo JSON RPC para criar, atualizar e buscar objetos Zabbix (como hosts, itens, gráficos e outros) ou executar quaisquer outras tarefas customizadas.

Zabbix Server

- um processo central do software Zabbix que executa monitoramento, interage com os Zabbix Proxies e Agents, calcula gatilhos, envia notificações; um repositório central de dados.

Zabbix Proxy

- um processo que pode coletar dados em nome do Zabbix Server, tirando um pouco da carga de processamento do Server.

Zabbix Agent

- um processo implementado nos alvos de monitoramento para monitorar ativamente os recursos e aplicações locais.

Zabbix Agent 2

- uma nova geração do Zabbix Agent para monitorar ativamente recursos e aplicações locais, permitindo usar plugins customizados para monitoramento.

Attention:

Pelo fato de o Zabbix Agent 2 compartilhar muita funcionalidade com o Zabbix Agent, o termo "Zabbix Agent" na documentação se aplica a ambos - Zabbix Agent e Zabbix Agent 2, se o comportamento funcional é o mesmo. O Zabbix Agent 2 é especificamente nomeado apenas se sua funcionalidade for diferente.

criptografia

- suporte de comunicações criptografadas entre componentes Zabbix (Server, Proxy, Agent, utilitários zabbix_sender e zabbix_get) usando o protocolo Transport Layer Security (TLS).

descoberta de rede

- descoberta automatizada de dispositivos de rede.

descoberta de baixo-nível

- descoberta automatizada de entidades de baixo-nível em um dispositivo em particular (p.e. sistemas de arquivo, interfaces de rede, etc.).

regra de descoberta de baixo-nível

- conjunto de definições para descoberta automatizada de entidades de baixo-nível em um dispositivo.

protótipo de item

- uma métrica com certos parâmetros como variáveis, pronta para descoberta de baixo-nível. Após a descoberta de baixo-nível as variáveis são automaticamente substituídas com os parâmetros reais descobertos e a métrica automaticamente inicia a coleta de dados.

protótipo de gatilho

- um gatilho com certos parâmetros como variáveis, pronto para descoberta de baixo-nível. Após a descoberta de baixo-nível as variáveis são automaticamente substituídas com os parâmetros reais descobertos e a métrica automaticamente inicia a coleta de dados.

Protótipos de algumas outras entidades do Zabbix estão também em uso na descoberta de baixo-nível - protótipos de gráfico, protótipos de host, protótipos de grupo de host.

autoregistro de agente

- processo automatizado através do qual um Zabbix Agent a si mesmo é registrado como um host e começa a monitorar.

3. Processos Zabbix

Por favor use a barra lateral para acessar o conteúdo na seção de processo do Zabbix.

2 Servidor

Visão geral

O Zabbix Server é o componente central da solução.

O servidor gerencia a coleta e recebimento de dados, calcula o estado das triggers, envia notificações aos usuários. Ele é o componente para o qual os agentes e proxies enviam dados sobre a disponibilidade, performance e integridade dos sistemas monitorados. O servidor também pode executar por si só verificações remotas nos dispositivos monitorados, estas verificações ocorrem quando se utiliza itens do tipo "verificação simples".

O servidor gerencia o repositório central de configuração, estatísticas e armazenamento de dados operacionais, é ele quem irá alertar os administradores quando os incidentes ocorrerem.

As funcionalidades básicas de uma solução de monitoração baseada em Zabbix é distribuída em três componentes: Zabbix Server, interface web e banco de dados (SGDB).

Todas as informações de configuração da monitoração são armazenadas no banco de dados, tanto o Servidor quanto a Interface Web do Zabbix interagem com o SGDB. Por exemplo, quando você utiliza a interface web (ou a API) para adicionar itens, eles são

salvos em uma tabela do SGDB. Em paralelo a isso o Zabbix Server, uma vez a cada minuto, irá buscar, na tabela de itens, a lista de itens que deverão ser monitorados. É por isso que pode demorar até dois minutos para que uma modificação feita na Interface Web comece a produzir efeitos na tela de dados recentes.

Processo do Servidor

O Zabbix Server é executado como um processo de segundo plano (daemon). O exemplo abaixo demonstra uma das formas de inicia-lo:

```
shell> cd sbin
shell> ./zabbix_server
```

Você pode utilizar alguns parâmetros com o Zabbix Server:

-c --config <arquivo>	caminho absoluto (completo) para o arquivo de configuração (o padrão é /etc/zabbix/zabbix_server.conf)
-R --runtime-control <opção>	executa funções administrativas
-h --help	apresenta o help de parâmetros
-V --version	apresenta o número de versão

Note:

O controle em tempo de execução não é suportado em OpenBSD e em NetBSD.

Exemplos de linha de comando com parâmetros:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf
shell> zabbix_server --help
shell> zabbix_server -V
```

Controle em tempo de execução

Opções do controle em tempo de execução:

Opção	Descrição	Objetivo
config_cache_reload	Atualiza o cache de configuração. O comando é ignorado se o cache já estiver atualizado.	
log_level_increase[=<alvo>]	Aumenta o nível de log, afeta todos os processos se o alvo não for especificado.	pid - Identificador do processo (1 a 65535) tipo do processo - Restringe a todos os processos de determinado tipo (Ex.: poller) tipo do processo,N - Restringe a determinado processo de um tipo específico (Ex.: poller,3)
log_level_decrease[=<alvo>]	Reduz o nível de log, afeta todos os processos se o alvo não for especificado.	

O PID do processo a se modificar o nível de log deverá estar entre 1 e 65535. Em ambientes com muitos processos a modificação poderá ser feita em um processo específico.

Exemplo de utilização do controle em tempo de execução para recarregar o cache de configuração do Zabbix Server:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R config_cache_reload
```

Exemplos de utilização do controle em tempo de execução para modificar o nível de log:

Aumenta o nível de log de todos os processos:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase
```

Aumenta o nível de log do segundo processo de pooler:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2
```

Aumenta o nível de log do processo com PID 1234:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234
```

Diminui o nível de log de todos os processos do pooler HTTP:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Inicie manualmente

Se o acima não funcionar você terá que iniciar manualmente. Encontre o caminho para o binário do zabbix_server e execute:

```
shell> zabbix_server
```

Você pode usar os seguintes parâmetros de linha de comando com o Zabbix server:

-c --config <file>	caminho para o arquivo de configuração (o padrão é /usr/local/etc/zabbix_server.conf)
-f --foreground	executa o Zabbix Server ao fundo (foreground)
-R --runtime-control <option>	realiza funções administrativas
-h --help	apresenta esta ajuda
-V --version	exibe número de versão

Exemplos de execução do Zabbix Server com parâmetros de linha de comando:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf
shell> zabbix_server --help
shell> zabbix_server -V
```

Controle em tempo de execução

Opções de controle em tempo de execução:

Opção	Descrição	Alvo
config_cache_reload	Recarrega o cache de configuração. Ignorado se o cache atualmente está sendo carregado.	
diaginfo[=<target>]	Reúne informações de diagnóstico no arquivo de log do Server.	historycache - estatísticas de cache de histórico valuecache - estatísticas de cache de valor preprocessing - estatísticas de gerenciados de pré-processamento alerting - estatísticas de gerenciador de alerta lld - estatísticas de gerenciador de LLD locks - lista de mutexes (é vazio em sistemas **BSD*)
ha_status	Registra (log) o estado do cluster de alta disponibilidade (HA).	
ha_remove_node=target	Remove o nó de alta disponibilidade (HA) especificado pelo seu número listado. Note que nós ativos/em espera não podem ser removidos.	target - número do nó na lista (pode ser obtido pela execução de ha_status)
ha_set_failover_delay=delay	Configura atraso de recuperação de falha (failover) de alta disponibilidade (HA). Sufixos de tempo são suportados, p.e. 10s, 1m.	
secrets_reload	Recarrega segredos do Vault.	
service_cache_reload	Recarrega o cache do gerenciador de serviço.	
snmp_cache_reload	Recarrega cache SNMP, limpa as propriedades SNMP (engine time, engine boots, engine id, credentials) para todos os hosts.	
housekeeper_execute	Inicia procedimento de housekeeping. Ignorado se o procedimento de housekeeping está atualmente em progresso.	
trigger_housekeeper_execute	Inicia o procedimento de gatilho de housekeeping. Ignorado se o procedimento de gatilho de housekeeping está atualmente em progresso.	

Opção	Descrição	Alvo
<code>log_level_increase[=<target>]</code>	Aumenta nível de log, afeta todos os processos se alvo não for especificado. Não suportado nos sistemas **BSD* .	process type - Todos os processos do tipo especificado (p.e., poller) Veja todos os tipos de processo do Server . process type,N - Tipo e número do processo (p.e., poller,3) pid - Identificador do processo (1 até 65535). Para valores maiores especifique alvo como 'process type,N'.
<code>log_level_decrease[=<target>]</code>	Diminui nível de log, afeta todos os processos se alvo não for especificado. Não suportado nos sistemas **BSD* .	

Exemplo de utilização de controle em tempo de execução para recarregar o cache de configuração do Server:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R config_cache_reload
```

Exemplo de utilização de controle em tempo de execução para reunir informação de diagnóstico:

Reúne todas as informações de diagnóstico disponíveis no arquivo de log do Server:

```
shell> zabbix_server -R diaginfo
```

Reúne estatísticas de cache de histórico no arquivo de log do Server:

```
shell> zabbix_server -R diaginfo=historycache
```

Exemplo de utilização de controle em tempo de execução para recarregar cache SNMP:

```
shell> zabbix_server -R snmp_cache_reload
```

Exemplo de utilização de controle em tempo de execução para execução de gatilho de housekeeper:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R housekeeper_execute
```

Exemplo de utilização de controle em tempo de execução para alterar nível de log:

Aumenta nível de log de todos os processos:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase
```

Aumenta o nível de log do segundo processo de poller:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=poller,2
```

Aumenta nível de log do processo com PID 1234:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_increase=1234
```

Diminui o nível de log de todos os processos de poller http:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf -R log_level_decrease="http poller"
```

Exemplo de configuração de atraso de failover HA para o mínimo de 10 segundos:

```
shell> zabbix_server -R ha_set_failover_delay=10s
```

Scripts de inicialização

Os scripts são utilizados para iniciar automaticamente os processos do Zabbix Server durante o processo de inicialização e finalização da máquina. Tais scripts podem ser localizados no diretório `misc/init.d` do código fonte da solução.

Plataformas suportadas

Devido aos requisitos de segurança e a natureza de missão crítica do funcionamento do Zabbix Server, o UNIX é o único sistema operacional que pode entregar de forma consistente o desempenho, tolerância a falhas e resiliência necessários. O Zabbix opera como uma das soluções líderes de mercado.

O Zabbix Server é testado nas seguintes plataformas:

- Linux
- Solaris

- AIX
- HP-UX
- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server
- Tru64/OSF1

Note:

O Zabbix pode funcionar em outros sistemas operacionais baseados no UNIX.

Start-up scripts

The scripts are used to automatically start/stop Zabbix processes during system's start-up/shutdown. The scripts are located under directory `misc/init.d`.

Server process types

- `alert_manager` - alert queue manager
- `alert_syncer` - alert DB writer
- `alerter` - process for sending notifications
- `availability_manager` - process for host availability updates
- `configuration_syncer` - process for managing in-memory cache of configuration data
- `discoverer` - process for discovery of devices
- `escalator` - process for escalation of actions
- `history_poller` - process for handling calculated, aggregated and internal checks requiring a database connection
- `history_syncer` - history DB writer
- `housekeeper` - process for removal of old historical data
- `http_poller` - web monitoring poller
- `icmp_pinger` - poller for icmping checks
- `ipmi_manager` - IPMI poller manager
- `ipmi_poller` - poller for IPMI checks
- `java_poller` - poller for Java checks
- `lld_manager` - manager process of low-level discovery tasks
- `lld_worker` - worker process of low-level discovery tasks
- `poller` - normal poller for passive checks
- `preprocessing_manager` - manager of preprocessing tasks
- `preprocessing_worker` - process for data preprocessing
- `problem_housekeeper` - process for removing problems of deleted triggers
- `proxy_poller` - poller for passive proxies
- `report_manager` - manager of scheduled report generation tasks
- `report_writer` - process for generating scheduled reports
- `self-monitoring` - process for collecting internal server statistics
- `snmp_trapper` - trapper for SNMP traps
- `task_manager` - process for remote execution of tasks requested by other components (e.g. close problem, acknowledge problem, check item value now, remote command functionality)
- `timer` - timer for processing maintenances
- `trapper` - trapper for active checks, traps, proxy communication
- `unreachable_poller` - poller for unreachable devices
- `vmware_collector` - VMware data collector responsible for data gathering from VMware services

The server log file can be used to observe these process types.

Various types of Zabbix server processes can be monitored using the **zabbix[process,<type>,<mode>,<state>]** internal **item**.

Supported platforms

Due to the security requirements and mission-critical nature of server operation, UNIX is the only operating system that can consistently deliver the necessary performance, fault tolerance and resilience. Zabbix operates on market leading versions.

Zabbix server is tested on the following platforms:

- Linux
- Solaris
- AIX
- HP-UX

- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server
- Tru64/OSF1

Note:

Zabbix may work on other Unix-like operating systems as well.

Locale

Note that the server requires a UTF-8 locale so that some textual items can be interpreted correctly. Most modern Unix-like systems have a UTF-8 locale as default, however, there are some systems where that may need to be set specifically.

1 Cluster de alta disponibilidade

Visão geral

O modo de alta disponibilidade oferece proteção contra falhas de software/hardware para o Zabbix Server e permite minimizar o tempo de parada durante manutenções de software/hardware.

O cluster de alta disponibilidade (HA) é uma solução opcional (opt-in) e é suportada para Zabbix Server. A solução de HA nativa é projetada para ser simples de usar, funcionará entre sites e não possui requisitos específicos para os bancos de dados que o Zabbix reconhece. Os usuários são livres para utilizar a solução de HA nativa do Zabbix, ou uma solução de HA de terceiros, dependendo do que melhor atende as necessidades de alta disponibilidade em seus ambientes.

A solução consiste de múltiplas instâncias ou nós do zabbix_server. Cada nó:

- é configurado separadamente (arquivo de configuração, scripts, criptografia, exportação de dados)
- usa o mesmo banco de dados
- possui vários modos: ativo, em espera, indisponível, parado

Apenas um nó pode estar ativo (trabalhando) por vez. Os nós em espera não fazem coleta de dados, processamento ou outras atividades regulares do Server; eles não esperam por comunicação nas portas; eles têm conexões mínimas com o banco de dados.

Ambos os nós ativos e em espera atualizam seus horários de último acesso a cada 5 segundos. Cada nó em espera monitora o horário do último acesso do nó ativo. Se horário do último acesso do nó ativo estiver acima de 'atraso de recuperação de falha' segundos, o nó em espera torna a si o nó ativo e associa o estado de 'indisponível' ao nó anteriormente ativo.

O nó ativo monitora sua própria conectividade de banco de dados - se for perdida por mais do que atraso de recuperação de falha - 5 segundos, ele deve parar todo o processamento e alterar para o modo de espera. O nó ativo também monitora o estado dos nós em espera - se o horário do último acesso de um nó em espera for maior do que 'atraso de recuperação de falha' segundos, o nó em espera recebe o estado de 'indisponível'.

O atraso de recuperação de falha é configurável, com o valor mínimo sendo de 10 segundos.

Os nós são projetados para serem compatíveis entre versões secundárias (minor) do Zabbix.

Habilitando cluster HA

Configurações do Server

Para transformar qualquer Zabbix Server de um servidor isolado (standalone) em um nó de cluster HA, especifique o parâmetro `HANodeName` (nome do nó) na **configuração** do Server.

O parâmetro de endereço do nó `NodeAddress` (endereço:porta), se configurado, deve ser usado pelo Frontend do nó ativo, sobrescrevendo o valor presente no `zabbix.conf.php`.

Preparing frontend

Make sure that Zabbix server address:port is **not defined** in the frontend configuration.

Zabbix frontend will autodetect the active node by reading settings from the nodes table in Zabbix database. Node address of the active node will be used as the Zabbix server address.

Configuração do Proxy

Para habilitar conexões à múltiplos Servers em um ambiente de alta disponibilidade, liste os endereços de nó de HA no **parâmetro** `Server` do Proxy, separados por ponto-e-vírgula.

Configurações do Agent

Para habilitar conexões à múltiplos Servers em um ambiente de alta disponibilidade, liste os endereços de nó de HA no **parâmetro** `ServerActive` do agente, separados por ponto-e-vírgula.

Failover to standby node

Zabbix will fail over to another node automatically if the active node stops. There must be at least one node in standby status for the failover to happen.

How fast will the failover be? All nodes update their last access time (and status, if it is changed) every 5 seconds. So:

- If the active node shuts down and manages to report its status as "shut down", another node will take over within **5 seconds**.
- If the active node shuts down/becomes unavailable without being able to update its status, standby nodes will wait for the **failover delay** + 5 seconds to take over

The failover delay is configurable, with the supported range between 10 seconds and 15 minutes (one minute by default). To change the failover delay, you may run:

```
zabbix_server -R ha_set_failover_delay=5m
```

Gerenciando um cluster HA

O estado atual do cluster HA pode ser gerenciado usando as opções de **controle em tempo de execução** dedicadas:

- `ha_status` - registra o estado do cluster HA no log do Zabbix Server;
- `ha_remove_node=alvo` - remove um nó HA identificado por seu <alvo> - número do nó na lista (o número pode ser obtido do resultado da execução de `ha_status`). Note que nós ativos/em espera não podem ser removidos.
- `ha_set_failover_delay=atraso` - configure o atraso de recuperação de falha de HA (sufixos de tempo são suportados, p.e. 10s, 1m)

O estado de um nó pode ser monitorado:

- em Relatórios → **Informação do sistema**
- no widget de dashboard **Informação de sistema**
- usando a opção de controle em tempo de execução `ha_status` do Server (veja acima).

O item interno `zabbix[cluster,discovery,nodes]` pode ser usado para descoberta de nó, pois ele retorna um JSON com informações de nós de alta disponibilidade.

Desabilitando um cluster HA

Para desabilitar um cluster de alta disponibilidade:

- faça cópia de backup dos arquivos de configuração
- pare os nós em espera
- remova o parâmetro `HANodeName` do servidor primário ativo
- reinicie o servidor primário (ele iniciará em modo isolado (standalone))

#####Detalhes da implementação

O cluster de alta disponibilidade (HA) é uma solução opcional e é suportada pelo Zabbix server. A solução nativa de HA está designada para ser simples de usar, funcionará em todos os sites e não tem requisitos específicos para o banco de dados que o Zabbix reconhece. Os usuário são livres para usar a solução de HA nativa do Zabbix, ou uma solução de HA de terceiros, dependendo do que irá atender melhor os requisitos de alta disponibilidade em seu ambiente.

A solução consiste de múltiplas instâncias do `zabbix_server` ou nós. Cada nó:

- é configurado separadamente
- usa o mesmo banco de dados
- pode ter vários modos: ativo, espera, indisponível, parado

Apenas um nó pode estar ativo (trabalhando) por vez. Um nó de espera executa apenas o processo gerenciador do HA. Um nó de espera não realiza a coleta de dados, processamento ou outras atividades regulares do servidor; eles não escutam nas portas; eles tem conexões mínimas com o banco de dados.

Ambos os nós ativos e em espera atualizam sua última hora de acesso a cada 5 segundos. Cada nó em espera monitora o último horário de acesso do nó ativo. Se a última hora de acesso do nó ativo for maior que o 'atraso de failover' em segundos, o nó de espera muda para ser o nó ativo e atribui o status de 'indisponível' para o nó ativo anteriormente.

O nó ativo monitora sua própria conectividade com o banco de dados - se ela for perdida por mais de 'failover delay -5' segundos, ele deve parar todos os processos e mudar para o modo de espera. O nó ativo também monitora o status dos nós em espera - se o último horário de acesso do nó de espera for maior que o 'failover delay' em segundos, o nó em espera muda o status para 'indisponível'.

3 Agent 2

Visão geral

O Zabbix Agent 2 é uma nova geração do agente Zabbix e pode ser usado no lugar do Zabbix Agent. O Zabbix Agent 2 foi desenvolvido para:

- reduzir o número de conexões TCP
- fornecer concorrência aprimorada de verificações
- ser facilmente extensível com plugins. Um plugin deve ser capaz de:
 - fornecer verificações triviais consistindo de apenas poucas linhas simples de código
 - fornecer verificações complexas consistindo de scripts de longa execução e aquisição de dados isolada com envio periódico dos dados de volta
- ser uma substituição opcional para o Zabbix Agent (em que ele suporta todas as funcionalidades anteriores)

O Agent 2 é escrito em Go (com algum código em C reutilizado do Zabbix Agent). Um ambiente Go configurado com uma [versão de Go](#) atualmente suportada é necessário para compilar o Zabbix Agent 2.

O Agent 2 não possui suporte nativo a execução com daemon (daemonization) no Linux; ele pode ser executado como um [serviço do Windows](#).

Verificações passivas funcionam de forma similar ao Zabbix Agent. As verificações ativas suportam intervalos agendados/flexíveis e concorrência de verificação em um Server ativo.

Concorrência de verificação

Verificações de diferentes plugins podem ser executadas de forma concorrente. O número de verificações concorrentes dentro de um plugin é limitada pela configuração de capacidade do plugin. Cada plugin pode ter uma configuração de capacidade fixada em código (sendo 100 o padrão) que pode ser diminuída usando a configuração `Plugins.<Plugin name>.Capacity=N` no [parâmetro](#) de configuração Plugins.

Veja também: [Orientações para desenvolvimento de plugin](#).

Plataformas suportadas

O Agent 2 é suportado pelas plataformas Linux e Windows.

Se instalando a partir dos pacotes, o Agent 2 é suportado em:

- RHEL/CentOS 6, 7, 8
- SLES 15 SP1+
- Debian 9, 10
- Ubuntu 18.04, 20.04

No Windows o Agent 2 é suportado em:

- Windows Server 2008 R2 e mais recente
- Windows 7 e mais recente

Instalação

O Zabbix Agent 2 está disponível em pacotes pré-compilados. Para compilar o Zabbix Agent 2 a partir dos fontes você tem que especificar a opção de configuração `--enable-agent2`.

Opções

Os seguintes parâmetros de linha de comando podem ser usados com o Zabbix Agent 2:

Parâmetro	Descrição
-c --config <arquivo-configuração>	Caminho para o arquivo de configuração. Você pode usar esta opção para informar um arquivo de configuração que não é o arquivo padrão. No UNIX, o padrão é <code>/usr/local/etc/zabbix_agent2.conf</code> ou como configurado pelas variáveis de tempo de compilação <code>--sysconfdir</code> ou <code>--prefix</code>
-f --foreground	Executa o agente ao fundo (foreground) (padrão: true (verdadeiro)).

Parâmetro	Descrição
-p --print	Apresenta itens conhecidos e sai. Nota: Para retornar os resultados de parâmetro do usuário também, você deve especificar o arquivo de configuração (caso ele não esteja na localização padrão).
-t --test <item key>	Testa o item especificado e sai. Nota: Para retornar os resultados de parâmetro do usuário também, você deve especificar o arquivo de configuração (caso ele não esteja na localização padrão).
-h --help	Apresenta informações de ajuda e sai.
-v --verbose	Apresenta informações de depuração (debugging). Use esta opção com as marcações -p e -t.
-V --version	Apresenta o número de versão do agente e sai.
-R --runtime-control <option>	Executa funções administrativas. Veja controle em tempo de execução .

Exemplos específicos de utilização de parâmetros de linha de comando:

- apresenta todos os itens nativos com valores
- testa um parâmetro de usuário com a chave "mysql.ping" definida no arquivo de configuração especificado

```
shell> zabbix_agent2 --print
```

```
shell> zabbix_agent2 -t "mysql.ping" -c /etc/zabbix/zabbix_agentd.conf
```

Controle em tempo de execução (runtime)

O controle em tempo de execução fornece algumas opções para controle remoto.

Opção	Descrição
log_level_increase	Aumenta nível de log.
log_level_decrease	Diminui nível de log.
metrics	Lista de métricas disponíveis.
version	Exibe versão do agente.
userparameter_reload	Recarrega parâmetros de usuário do arquivo de configuração atual. Note que o UserParameter é única opção de configuração do agente que será recarregada.
help	Exibe informações de ajuda sobre controle em tempo de execução.

Exemplos:

- aumentando o nível de log para o Agent 2
- apresenta opções de controle em tempo de execução

```
shell> zabbix_agent2 -R log_level_increase
```

```
shell> zabbix_agent2 -R help
```

Arquivo de configuração

Os parâmetros de configuração do Agent 2 são predominantemente compatíveis com o Zabbix Agent com algumas exceções.

Novos parâmetros	Descrição
ControlSocket	O caminho para o socket de controle em tempo de execução. O Agent 2 usa um socket de controle para comandos em tempo de execução .
EnablePersistentBuffer,	Estes parâmetros são usados para configurar o armazenamento persistente no Agent 2 para itens ativos. Plugins podem ter seus próprios parâmetros, no formato <code>Plugins.<Nome do plugin>.<Parâmetro>=<valor></code> . Um parâmetro de plugin comum é Capacity (capacidade), configurando o limite de verificações que podem ser executadas ao mesmo tempo.
PersistentBufferFile,	
PersistentBufferPeriod	
Plugins	

Novos parâmetros	Descrição
StatusPort	A porta na qual o Agent 2 aguardará comunicação (listening) para requisições de estado HTTP e exibição de uma lista de plugins configurados e alguns parâmetros internos
Parâmetros descontinuados	Descrição
AllowRoot, User	Não suportado porque operação como daemon (daemonization) não é suportada.
LoadModule, LoadModulePath	Módulos carregáveis não são suportados.
StartAgents	Este parâmetro era usado no Zabbix Agent para aumentar a concorrência de verificações passivas ou desabilitá-los. No Agent 2, a concorrência é configurada no nível do plugin e pode ser limitada por uma configuração de capacidade. Considerando que desabilitar verificações passivas não é atualmente suportado.
HostInterface, HostInterfaceItem	Não suportado ainda.

Para mais detalhes veja as opções do arquivo de configuração para [zabbix_agent2](#).

Códigos de saída

A partir da versão 4.4.8 o Zabbix Agent 2 também pode ser compilado com versões mais antigas do OpenSSL (1.0.1, 1.0.2).

Neste caso o Zabbix oferece mutexes para travamento no OpenSSL. Se um travamento ou destravamento mutex falhar então uma mensagem de erro é apresentada na saída de erro padrão (STDERR) e o Agent 2 sai com código de erro 2 ou 3, respectivamente.

3 Agente

Visão geral

O agente Zabbix é instalado no dispositivo alvo da monitoração. Possui capacidade de monitorar de monitorar ativamente os recursos e aplicações locais (discos e partições, memória, estatísticas do processador, etc).

O agente concentra as informações locais sobre o dispositivo monitorado para posterior envio ao servidor ou proxy Zabbix (dependendo da configuração). Em caso de falhas (como um disco cheio ou a interrupção de um processo) o servidor Zabbix pode alertar ativamente os administradores do ambiente sobre o ocorrido.

Os agentes Zabbix são extremamente eficientes pois utilizam chamadas nativas do sistema operacional para obter as informações estatísticas.

Verificações passivas e ativas

Os agentes Zabbix podem executar verificações passivas ou ativas.

Em uma **verificação passiva** o agente responde à uma requisição de informações. O servidor ou o proxy Zabbix requisitam o dado toda vez que é necessário (uso de CPU, memória, disco, etc), o agente responde com o resultado do teste solicitado.

O processo de **verificação ativa** requer um processamento mais complexo. O agente precisa primeiro receber a lista de itens a monitorar e o intervalo entre coletas pretendido. Esta informação vem do servidor Zabbix através de requisições periódicas do agente.

A verificação ativa permite que o agente continue executando o perfil de monitoração mesmo quando o servidor Zabbix está indisponível, enviando posteriormente e de forma retroativa o resultado dos testes.

A definição se a verificação deve ocorrer de forma passiva ou ativa é configurada através do **tipo do item**, na interface web do Zabbix. Um agente Zabbix pode processar itens do tipo 'Agente Zabbix' ou "Agente Zabbix (ativo)".

Plataformas suportadas

O agente Zabbix é suportado por:

- Linux
- IBM AIX
- FreeBSD
- NetBSD
- OpenBSD

- HP-UX
- Mac OS X
- Solaris: 9, 10, 11
- Windows: 2000, Server 2003, XP, Vista, Server 2008, 7

Instalação

Veja as [instruções de instalação](#) para o agente.

Attention:

Em geral os agentes 32bits do Zabbix conseguirão ser executados em ambientes 64bits, mas em alguns casos poderá ocorrer falha.

Instalação

Veja [instalação de pacotes](#) local para instruções sobre como instalar o agente Zabbix como pacote.

Veja instruções alternativas para [instalação manual](#) caso não queira instalar via pacotes.

::: Aviso importante Em geral, Agentes Zabbix 32bit irão funcionar com sistemas 64bit , mas podem falhar em alguns casos. :::

Controle em tempo de execução

Opções de controle em tempo de execução:

Opção	Descrição	Alvo
log_level_increase[=<alvo>]	Aumenta o nível de log, afeta todos os processos se o alvo não for especificado.	pid - Identificador do processo (1 a 65535) tipo do processo - Restringe a todos os processos de determinado tipo (Ex.: poller) tipo do processo,N - Restringe a determinado processo de um tipo específico (Ex.: poller,3)
log_level_decrease[=<alvo>]	Reduz o nível de log, afeta todos os processos se o alvo não for especificado.	

O PID do processo a se modificar o nível de log deverá estar entre 1 e 65535. Em ambientes com muitos processos a modificação poderá ser feita em um processo específico.

Exemplo de utilização do controle em tempo de execução para modificar o nível de log:

Increase log level of all processes:

```
shell> zabbix_agentd -c /usr/local/etc/zabbix_agentd.conf -R log_level_increase
```

Aumenta o nível de log do segundo processo do ouvinte (listener):

```
shell> zabbix_agentd -c /usr/local/etc/zabbix_agentd.conf -R log_level_increase=listener,2
```

Aumenta o nível de log do processo com PID 1234:

```
shell> zabbix_agentd -c /usr/local/etc/zabbix_agentd.conf -R log_level_increase=1234
```

Reduz o nível de log de todas os processos de verificação ativa:

```
shell> zabbix_agentd -c /usr/local/etc/zabbix_agentd.conf -R log_level_decrease="active checks"
```

Processo de usuário

O agente Zabbix foi desenhado para ser executado como um processo “não-root”. Ele pode ser executado com a permissão do usuário que o iniciou. Neste cenário ele irá executar sem nenhum problema.

Se você tentar inicia-lo com o usuário ‘root’, ele alternará seu permissionamento de execução para o usuário ‘zabbix’, que deverá existir em seu ambiente. Você só poderá rodar o Servidor Zabbix como ‘root’ se modificar o parâmetro ‘AllowRoot’ no arquivo de configuração.

Agent em sistemas Windows

O Zabbix Agent no Windows é executado como um Serviço do Windows.

Executando o agente em ambiente Microsoft Windows

Veja o manual do [agente no Windows](#) para detalhes sobre como instalar, configurar e executar o agente neste sistema operacional.

Sintaxe de linha de comando do agente no Windows:

```
zabbix_agentd.exe [-c arquivo-de-configuração]
zabbix_agentd.exe [-c arquivo-de-configuração] -p
zabbix_agentd.exe [-c arquivo-de-configuração] -t chave-do-item
zabbix_agentd.exe [-c arquivo-de-configuração] -i [-m]
zabbix_agentd.exe [-c arquivo-de-configuração] -d [-m]
zabbix_agentd.exe [-c arquivo-de-configuração] -s [-m]
zabbix_agentd.exe [-c arquivo-de-configuração] -x [-m]
zabbix_agentd.exe -h
zabbix_agentd.exe -V
```

Os parâmetros a seguir podem ser utilizados.

Options:

```
-c --config <arquivo>          caminho absoluto (completo) para o arquivo de configuração (o padrão é c:\zabbix\zabbix_agentd.conf)
-h --help                      apresenta o help de parâmetros
-V --version                   apresenta o número de versão
-p --print                     apresenta todos os itens (chaves) possíveis
-t --test <chave do item>     testa um item específico e retorna o resultado
```

Functions:

```
-i --install                   Instala o serviço do agente Zabbix
-d --uninstall                Desinstala o serviço do agente Zabbix
-s --start                    Inicia o serviço do agente Zabbix
-x --stop                     Finaliza o serviço do agente Zabbix
-m --multiple-agents          Nome do serviço com o hostname
```

Arquivo de configuração

Veja o manual do [arquivo de configuração](#) para detalhes de opções de configuração do agente Zabbix no Windows.

Códigos de saída

Antes da versão 2.2 do Zabbix o agente retornava 0 em caso de sucesso e 255 em caso de falha. A partir desta versão o agente passou a retornar 0 para sucesso e 1 para falha.

Runtime control

With runtime control options you may change the log level of agent processes.

Option	Description	Target
log_level_increase[=<target>]	Increase log level. If target is not specified, all processes are affected.	Target can be specified as: process type - all processes of specified type (e.g., listener) See all agent process types . process type,N - process type and number (e.g., listener,3) pid - process identifier (1 to 65535). For larger values specify target as 'process-type,N'.
log_level_decrease[=<target>]	Decrease log level. If target is not specified, all processes are affected.	
userparameter_reload	Reload user parameters from the current configuration file. Note that UserParameter is the only agent configuration option that will be reloaded.	

Examples:

- increasing log level of all processes
- increasing log level of the third listener process
- increasing log level of process with PID 1234
- decreasing log level of all active check processes

```
shell> zabbix_agentd -R log_level_increase
shell> zabbix_agentd -R log_level_increase=listener,3
shell> zabbix_agentd -R log_level_increase=1234
shell> zabbix_agentd -R log_level_decrease="active checks"
```

Note:

Runtime control is not supported on OpenBSD, NetBSD and Windows.

Agent process types

- `active checks` - process for performing active checks
- `collector` - process for data collection
- `listener` - process for listening to passive checks

The agent log file can be used to observe these process types.

Process user

Zabbix agent on UNIX is designed to run as a non-root user. It will run as whatever non-root user it is started as. So you can run agent as any non-root user without any issues.

If you will try to run it as 'root', it will switch to a hardcoded 'zabbix' user, which must be present on your system. You can only run agent as 'root' if you modify the 'AllowRoot' parameter in the agent configuration file accordingly.

Configuration file

For details on configuring Zabbix agent see the configuration file options for `zabbix_agentd` or `Windows agent`.

Locale

Note that the agent requires a UTF-8 locale so that some textual agent items can return the expected content. Most modern Unix-like systems have a UTF-8 locale as default, however, there are some systems where that may need to be set specifically.

Exit code

Before version 2.2 Zabbix agent returned 0 in case of successful exit and 255 in case of failure. Starting from version 2.2 and higher Zabbix agent returns 0 in case of successful exit and 1 in case of failure.

4 Proxy

Visão geral

O Zabbix Proxy é um processo que pode receber dados de um ou mais dispositivos monitorados e enviar ao Zabbix Server, basicamente ele funciona em nome do Zabbix Server (na visão do agente monitorado o Proxy passa a ser o Zabbix Server). Todos os dados recebidos são armazenados temporariamente (bufferizados), transferidos ao Zabbix Server que o Zabbix Proxy pertencer, sendo excluídos na sequência do armazenamento temporário do Proxy.

A utilização deste componente é opcional, mas normalmente é muito benéfica pois distribui a carga de monitoração normalmente atribuída ao Zabbix Server. Se toda a coleta de dados for feita através de Proxies o uso de CPU e de I/O no servidor responsável pelo Zabbix Server reduz significativamente.

O Zabbix Proxy é a solução ideal para a monitoração centralizada de localidades geograficamente dispersas e para redes gerenciadas remotamente.

O Zabbix Proxy requer um banco de dados em separado (normalmente um SQLite).

Attention:

Observe que o Proxy suporta SQLite, MySQL e PostgreSQL. O uso de Oracle ou IBM DB2 neste componente é uma escolha com riscos e limitações seus, exemplos podem ser encontrados em **retorno de valores** regras de autobusca.

Veja também: [Usando Proxies em ambientes distribuídos](#)

Processo do Proxy

O Zabbix Proxy é executado como um processo de background (Daemon). O proxy pode ser iniciado ao executar:

```
shell> cd sbin
shell> ./zabbix_proxy
```

Você pode utilizar alguns parâmetros com o Zabbix Proxy:

-c --config <arquivo>	caminho absoluto (completo) para o arquivo de configuração (o padrão é /etc/zabbix/zabbix_proxy.conf)
-R --runtime-control <opção>	executa funções administrativas
-h --help	apresenta o help de parâmetros
-V --version	apresenta o número de versão

Note:

O controle em tempo de execução não é suportado em OpenBSD e em NetBSD.

Exemplos de linha de comando com parâmetros:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf
shell> zabbix_proxy --help
shell> zabbix_proxy -V
```

Se instalado como pacote

Zabbix proxy roda como um processo em segundo plano. O proxy pode ser iniciado executando: `service zabbix-proxy start`

Isso funcionará na maioria dos sistemas GNU/Linux. Em alguns sistemas você pode necessitar executar:

```
/etc/init.d/zabbix-proxy start
```

De forma semelhante, para parar/reiniciar/ver status do Zabbix proxy, utilize os seguintes comandos:

```
service zabbix-proxy stop service zabbix-proxy restart service zabbix-proxy status
```

Processo de usuário

O Zabbix Proxy foi desenhado para ser executado como um processo "não-root". Ele pode ser executado com a permissão do usuário que o iniciou. Neste cenário ele irá executar sem nenhum problema.

Se você tentar inicia-lo com o usuário 'root', ele irá alternar seu permissionamento de execução para o usuário 'zabbix', que deverá **existir** em seu ambiente. Você só poderá rodar o Zabbix Proxy como 'root' se modificar o parâmetro 'AllowRoot' no arquivo de configuração.

Arquivo de configuração

Veja as opções do [arquivo de configuração](#) para detalhes sobre sua configuração.

Scripts de inicialização

Os scripts são utilizados para iniciar automaticamente os processos do Zabbix Proxy durante o processo de inicialização e finalização da máquina. Atualmente (29/11/15) estes scripts não vem junto com o código fonte da solução mas são facilmente obtidos através da cópia e edição dos scripts do Zabbix Server localizados no diretório misc/init.d do código fonte da solução.

Plataformas suportadas

Devido aos requisitos de segurança e a natureza de missão crítica do funcionamento do Zabbix Proxy, o UNIX é o único sistema operacional que pode entregar de forma consistente o desempenho, tolerância a falhas e resiliência necessários. O Zabbix opera como uma das soluções líderes de mercado.

O Zabbix Proxy é testado nas seguintes plataformas:

- Linux
- Solaris
- AIX
- HP-UX
- Mac OS X
- FreeBSD
- OpenBSD
- NetBSD
- SCO Open Server
- Tru64/OSF1

Note:

O Zabbix pode funcionar em outros sistemas operacionais baseados no UNIX.

Proxy process types

- `availability manager` - process for host availability updates
- `configuration syncer` - process for managing in-memory cache of configuration data
- `data sender` - proxy data sender
- `discoverer` - process for discovery of devices
- `heartbeat sender` - proxy heartbeat sender
- `history poller` - process for handling calculated, aggregated and internal checks requiring a database connection
- `history syncer` - history DB writer
- `housekeeper` - process for removal of old historical data
- `http poller` - web monitoring poller
- `icmp pinger` - poller for icmping checks
- `ipmi manager` - IPMI poller manager
- `ipmi poller` - poller for IPMI checks
- `java poller` - poller for Java checks
- `poller` - normal poller for passive checks
- `preprocessing manager` - manager of preprocessing tasks
- `preprocessing worker` - process for data preprocessing
- `self-monitoring` - process for collecting internal server statistics
- `snmp trapper` - trapper for SNMP traps
- `task manager` - process for remote execution of tasks requested by other components (e.g. close problem, acknowledge problem, check item value now, remote command functionality)
- `trapper` - trapper for active checks, traps, proxy communication
- `unreachable poller` - poller for unreachable devices
- `vmware collector` - VMware data collector responsible for data gathering from VMware services

The proxy log file can be used to observe these process types.

Various types of Zabbix proxy processes can be monitored using the **zabbix[process,<type>,<mode>,<state>]** internal **item**.

Supported platforms

Zabbix proxy runs on the same list of **server#supported platforms** as Zabbix server.

Locale

Note that the proxy requires a UTF-8 locale so that some textual items can be interpreted correctly. Most modern Unix-like systems have a UTF-8 locale as default, however, there are some systems where that may need to be set specifically.

5 Java gateway

Visão geral

O suporte nativo para monitoramento de aplicativos JMX existe na forma de um daemon chamado "Zabbix Java gateway, disponível desde a versão Zabbix 2.0. O Zabbix Java gateway é um daemon escrito em Java. Para saber o valor de um contador JMX específico em um Host, o servidor Zabbix consulta o Zabbix Java gateway, que utiliza **JMX management API** para consultar remotamente a aplicação de interesse. A aplicação não necessita de nenhum software extra instalado, apenas precisa ser iniciada com a opção `Dcom.sun.management.jmxremote` na linha de comando.

O Java gateway aceita conexão de entrada a partir do servidor ou proxy do Zabbix e apenas pode ser utilizado como um "passive proxy". Ao contrário do proxy Zabbix, também pode ser usado a partir do proxy Zabbix (os proxies do Zabbix não podem ser encadeados). O acesso a cada Java gateway é configurado diretamente no servidor Zabbix ou no arquivo de configuração do proxy. Assim, apenas um Java gateway pode ser configurado por vez no servidor ou no proxy Zabbix. Se um host tiver itens do tipo **JMX agent** e itens de outro tipo, somente os itens **JMX agent** serão encaminhados para recuperação de Java gateway.

Quando um item precisa ser atualizado por meio do Java gateway, o servidor ou proxy do Zabbix se conectará com o Java gateway e solicitará o valor, que o Java gateway por sua vez recupera e retorna para o servidor ou para o proxy. Portanto, o Java gateway não armazena nenhum valor em cache.

O servidor ou proxy do Zabbix tem um tipo específico de processos que se conectam ao Java gateway, controlados pela opção **StartJavaPollers**. Internamente, o Java gateway inicia várias threads, controladas pela opção **START_POLLERS option**. No servidor, se uma conexão levar mais que **Timeout** de segundos, será finalizada. Porém, o Java gateway ainda pode estar ocupado,

recuperando valores do contador JMX. Para resolver isso, existe a opção **TIMEOUT** no Java gateway, que permite configurar timeouts para operações de rede JMX.

O servidor ou proxy do Zabbix tentará agrupar as solicitações de pool para um único alvo JMX o máximo possível (afetado pelos intervalos de itens) e enviará essas solicitações para o Java gateway em uma única conexão para melhorar o desempenho. É sugerido que o número de **StartJavaPollers** seja menor ou igual a **START_POLLERS**. Caso contrário, pode haver situações em que nenhuma thread estará disponível no Java gateway para atender às solicitações recebidas; nesse caso, o Java gateway utiliza `ThreadPoolExecutor.CallerRunsPolicy`, significando que a thread principal atenderá a solicitação recebida e não aceitará novas solicitações temporariamente.

If you are trying to monitor Wildfly-based Java applications with Zabbix Java gateway, please install the latest jboss-client.jar available on the [Wildfly download page](#).

Se você estiver tentando monitorar aplicativos Wildfly-based Java com Java gateway do Zabbix, por favor, instale a última versão de jboss-client.jar disponível na página de downloads [Wildfly download page](#).

Obtendo o gateway Java

Você pode instalar o gateway Java a partir das fontes ou pacotes baixados do [Zabbix website](#).

Usando os links abaixo, você pode acessar informações sobre como obter e executar Zabbix Java gateway, como configurar o servidor Zabbix (ou proxy Zabbix) para utilizar Zabbix Java gateway para monitoramento JMX, e como configurar os itens no frontend Zabbix que correspondem a contadores JMX específicos .

Instalação a partir de	Instruções	Instruções
Fontes	Instalação	Configuração
pacotes RHEL/CentOS	Instalação	Configuração
pacotes Debian/Ubuntu	Instalação	Configuração

1 Setup a partir de fontes

Visão Geral

Se instalado **installed** a partir de fontes, as seguintes informações irão ajudá-lo a configurar o Zabbix **Java gateway**.

Visão Geral dos Arquivos

Se você obteve o Java gateway a partir das fontes, você deve ter acabado com uma coleção de scripts shell, JAR e arquivos de configuração em `$PREFIX/sbin/zabbix_java`. O papel desses arquivos é resumido abaixo.

`bin/zabbix-java-gateway-$VERSION.jar`

O próprio arquivo JAR do Java gateway. `lib/logback-core-0.9.27.jar` `lib/logback-classic-0.9.27.jar` `lib/slf4j-api-1.6.1.jar` `lib/android-json-4.3_r3.1.jar`

Dependências do Java gateway: Bibliotecas [Logback](#), [SLF4J](#), e [Android JSON](#).

`lib/logback.xml`

`lib/logback-console.xml`

Arquivos de configuração para Logback.

`shutdown.sh`

`startup.sh`

Scripts de conveniência para iniciar e parar Java gateway.

`settings.sh`

Arquivo de configuração que é fonte para os scripts de inicialização e desligamento acima.

Configurando e executando o Java gateway

Por padrão, o Java gateway escuta na porta 10052. Se você planeja executar o Java gateway em uma porta diferente, você pode especificar isso no script `settings.sh`. Consulte a descrição do arquivo de configuração do Java gateway [Java gateway configuration file](#) para especificar esta e outras opções.

Warning:

A porta Port 10052 não está registrada [IANA registered](#).

Uma vez que você esteja confortável com as configurações, pode iniciar o Java gateway executando o script de inicialização:

```
./startup.sh
```

Da mesma forma, uma vez que você não precise mais do Java gateway, execute o script de desligamento para pará-lo:

```
./shutdown.sh
```

Observe que, ao contrário do servidor ou do proxy, o Java gateway é leve e não precisa de um banco de dados.

Configurando os servidor para usar o Java gateway

Com o Java gateway funcionando, você precisa informar ao servidor Zabbix onde encontrar o Java gateway do Zabbix. Isso é feito especificando os parâmetros `JavaGateway` e `JavaGatewayPort` no arquivo de configuração do servidor [server configuration file](#). Se o host no qual a aplicação JMX está sendo executada estiver sendo monitorado por um proxy Zabbix, então você especifica os parâmetros de conexão no arquivo de configuração do proxy [proxy configuration file](#).

```
JavaGateway=192.168.3.14
```

```
JavaGatewayPort=10052
```

Por padrão, o servidor não inicia nenhum processo relacionado ao monitoramento JMX. Se você deseja usá-lo, no entanto, você precisa especificar o número de instâncias predefinidas dos pollers Java. Você faz isso da mesma forma que especifica os pollers e trappers regulares.

```
StartJavaPollers=5
```

Não se esqueça de reiniciar o servidor ou o proxy assim que terminar de configurá-los.

Debugando Java gateway

Caso haja algum problema com o Java gateway ou uma mensagem de erro que você vê sobre um item no frontend não seja descritiva o suficiente, você pode dar uma olhada no arquivo de log do Java gateway.

Por padrão, o Java gateway registra suas atividades no arquivo `/tmp/zabbix_java.log` com nível de log "info". Às vezes, essa informação não é suficiente e há necessidade de informações no nível de log "debug". Para aumentar o nível de registro, modifique o arquivo `lib/logback.xml` e altere o atributo de nível da tag `<root>` para "debug":

```
<root level="debug">
  <appender-ref ref="FILE" />
</root>
```

Observe que, ao contrário do servidor ou do proxy Zabbix, não é necessário reiniciar o Java gateway do Zabbix após alterar o arquivo `logback.xml` - as alterações no `logback.xml` serão automaticamente reconhecidas. Quando terminar a debugging, você pode retornar o nível de log para "info".

Se desejar registrar em um arquivo diferente ou em um meio completamente diferente, como um banco de dados, ajuste o arquivo `logback.xml` para atender às suas necessidades. Para mais detalhes, consulte o manual [Logback Manual](#).

Às vezes, para fins de debugging, é útil iniciar o Java gateway como um aplicativo de console em vez de um daemon. Para fazer isso, comente a variável `PID_FILE` em `settings.sh`. Se o `PID_FILE` for omitido, o script `startup.sh` inicia o Java gateway como um aplicativo de console e faz com que o Logback use o arquivo `lib/logback-console.xml`, que não apenas registra no console, mas também tem o nível de log "debug" ativado.

Finalmente, observe que, como o Java gateway usa o SLF4J para registro, você pode substituir o Logback pelo framework de sua escolha colocando um arquivo JAR apropriado no diretório `lib`. Para mais detalhes, consulte o manual [SLF4J Manual](#).

Monitoramento JMX

Para mais detalhes, consulte a página [JMX monitoring](#).

2 Configuração a partir de pacotes RHEL

Visão geral

Se você tiver instalado [installed](#) a partir de pacotes RHEL, a informação seguinte te ajudará na configuração do Zabbix [Java gateway](#).

Configurar e executar Java gateway

Os parâmetros de configuração do Java gateway do Zabbix podem ser ajustados no arquivo:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Para mais detalhes, ver configuração do Java gateway do Zabbix [parameters](#).

Para iniciar o Java gateway do Zabbix:

```
service zabbix-java-gateway restart
```

Para iniciar o Java gateway do Zabbix automaticamente na inicialização:

RHEL 7 e depois:

```
systemctl enable zabbix-java-gateway
```

RHEL anterior à versão 7:

```
chkconfig --level 12345 zabbix-java-gateway on
```

Configurar o servidor para uso com Java gateway

Com o Java gateway instalado e rodando, você precisa informar ao servidor Zabbix onde encontrar o Java gateway Zabbix. Isso é feito especificando os parâmetros `JavaGateway` e `JavaGatewayPort` na configuração do servidor [server configuration file](#). Se o host na qual a aplicação JMX estiver em execução e estiver sendo monitorado pelo Zabbix proxy, então você deve especificar os parâmetros de conexão na configuração [proxy configuration file](#).

```
JavaGateway=192.168.3.14
```

```
JavaGatewayPort=10052
```

Por padrão, o servidor não inicia nenhum processo relacionado ao monitoramento JMX. Contudo, caso você deseje usá-lo, você deve especificar o número de instâncias pre-forked do Java pollers. Você faz isso da mesma forma que você especifica os pollers e trappers regulares.

```
StartJavaPollers=5
```

Não esqueça de reiniciar o servidor ou proxy depois de terminar de configurá-los.

Debugar Java gateway

O arquivo de registro do Zabbix Java gateway é:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Se você deseja aumentar o registro, edite o arquivo:

```
/etc/zabbix/zabbix_java_gateway_logback.xml
```

e mude `level="info"` para `"debug"` ou até mesmo `"trace"` (para solução detalhada):

```
<configuration scan="true" scanPeriod="15 seconds">
```

```
[...]
```

```
  <root level="info">
```

```
    <appender-ref ref="FILE" />
```

```
  </root>
```

```
</configuration>
```

Monitorar JMX

Ver página [JMX monitoring](#) para mais detalhes.

3 Configuração a partir dos pacotes Debian/Ubuntu

Visão geral

Ao instalar a partir dos pacotes Debian/Ubuntu [installed](#), a informação seguinte ajudará na configuração do Java gateway do Zabbix [Java gateway](#).

Configurar e executar o Java gateway

A configuração do Java gateway pode ser ajustada no arquivo:

```
/etc/zabbix/zabbix_java_gateway.conf
```

Para mais detalhes, consulte a configuração do Zabbix Java gateway [parameters](#).

Para iniciar o Zabbix Java gateway:

```
service zabbix-java-gateway restart
```

Para iniciar automaticamente o Zabbix Java gateway on boot:

```
systemctl enable zabbix-java-gateway
```

Configurar o servidor para uso com Java gateway

Com o Java gateway instalado e rodando, você precisa informar ao servidor do Zabbix onde encontrar o Java gateway Zabbix. Isso é feito especificando os parâmetros `JavaGateway` e `JavaGatewayPort` na configuração do servidor [server configuration file](#). Se o host na qual a aplicação JMX estiver em execução e estiver sendo monitorado pelo Zabbix proxy, então você deve especificar os parâmetros de conexão na configuração do proxy [proxy configuration file](#).

```
JavaGateway=192.168.3.14
JavaGatewayPort=10052
```

Por padrão, o servidor não inicia nenhum processo relacionado ao monitoramento JMX. Contudo, caso você deseje usá-lo, você deve especificar o número de instâncias pre-forked do Java pollers. Você faz isso da mesma forma que você especifica os pollers e trappers regulares.

```
StartJavaPollers=5
```

Não esqueça de reiniciar o servidor ou proxy depois de terminar de configurá-los.

Debugar o Java gateway

O arquivo de log do Zabbix Java gateway é:

```
/var/log/zabbix/zabbix_java_gateway.log
```

Se você deseja aumentar o logging, edite o arquivo: `/etc/zabbix/zabbix_java_gateway_logback.xml`

e altere `level="info"` para `"debug"` ou até mesmo para `"trace"` (para solução de problemas mais detalhada):

```
<configuration scan="true" scanPeriod="15 seconds">
[...]
  <root level="info">
    <appender-ref ref="FILE" />
  </root>

</configuration>
```

Monitorar JMX

Consultar página [JMX monitoring](#) para mais detalhes.

6 Sender

Visão geral

O Zabbix Sender é um utilitário de linha de comando que pode ser usado para enviar dados de performance ao Zabbix Server para processamento.

O utilitário é comumente usado em scripts de usuário de longa execução para envio periódico de dados de disponibilidade e performance.

Para envio de resultados diretamente para o Zabbix Server ou Proxy, um [item do tipo gatilho](#) deve ser configurado.

Executando o Zabbix Sender

Um exemplo de execução do Zabbix Sender no UNIX:

```
shell> cd bin
shell> ./zabbix_sender -z zabbix -s "Linux DB3" -k db.connections -o 43
```

onde:

- `z` - host do Zabbix Server (endereço IP também pode ser usado)
- `s` - nome técnico do host monitorado (como registrado no Zabbix Frontend (interface web))
- `k` - chave de item
- `o` - valor para envio

Attention:

Opções que contêm espaços em branco, devem ser quotadas com aspas duplas.

O Zabbix Sender pode ser usado para enviar múltiplos valores a partir de um arquivo de entrada. Consulte a [página principal do Zabbix Sender](#) para mais informações.

Se um arquivo de configuração é especificado, o Zabbix Sender usa todos os endereços definidos no parâmetro de configuração ServerActive do agente para o envio de dados. Se o envio para um destes endereços falhar, o Sender tenta enviar para os outros endereços. Se o envio de um conjunto de dados falhar para um endereço, o conjunto seguinte não é enviado para este endereço.

O Zabbix Sender aceita strings com codificação UTF-8 (para ambos os sistemas baseados em UNIX e Windows) sem 'byte order mark (BOM)' em primeiro no arquivo.

Zabbix Sender no Windows pode ser executado de forma similar:

```
zabbix_sender.exe [options]
```

Desde o Zabbix 1.8.4, os cenários de envio em tempo real do zabbix_sender foram aperfeiçoados para reunir múltiplos valores passados a ele em sucessão próxima e enviá-los para o Server em uma única conexão. Um valor que não esteja distante mais do que 0.2 segundos do valor anterior pode ser colocado na mesma pilha, mas o tempo máximo de agrupamento (pooling) ainda é de 1 segundo.

Note:

O Zabbix Sender terminará se um parâmetro inválido (não seguindo a notação parâmetro=valor) estiver presente no arquivos de configuração especificado.

7 Get

Visão geral

O Zabbix Get é um utilitário de linha de comando que pode ser usado para se comunicar com o Zabbix Agent e recuperar uma informação requerida do agente.

Este utilitário é comumente utilizado para resolução de problemas nos Zabbix Agents.

Execução do Zabbix Get

Um exemplo de execução do Zabbix Get em UNIX para obter do agente o valor de carga de processador:

```
shell> cd bin
shell> ./zabbix_get -s 127.0.0.1 -p 10050 -k system.cpu.load[all,avg1]
```

Um outro exemplo de execução do Zabbix Get para captura de uma string de um website:

```
shell> cd bin
shell> ./zabbix_get -s 192.168.1.1 -p 10050 -k "web.page.regex[www.example.com,,,\"USA: ([a-zA-Z0-9.-]+)\"]"
```

Note que a chave de item aqui contém um espaço então aspas são usadas para destacar a chave do item no shell. As aspas não fazem parte da chave do item; Elas serão removidas pelo shell e não será passadas para o Zabbix Agent.

O Zabbix Get aceita os seguintes parâmetros de linha de comando:

-s --host <nome do host ou IP>	Especifica nome de host ou endereço IP de um host.
-p --port <número de porta>	Especifica o número de porta do agente em execução no host. Padrão é 10050.
-I --source-address <endereço IP>	Especifica endereço IP de origem.
-t --timeout <segundos>	Especifica limite de tempo (timeout). Intervalo válido: 1-30 segundos.
-k --key <chave do item>	Especifica chave de item da qual obter valor.
-h --help	Apresenta esta ajuda.
-V --version	Exibe número da versão.

Veja também a [página principal do Zabbix Get](#) para mais informações.

O Zabbix Get no Windows pode ser executado de forma similar a:

```
zabbix_get.exe [options]
```

8 JS

Visão geral

O zabbix_js é um utilitário de linha de comando que pode ser usado para teste de embedded script.

Este utilitário executará um script de usuário com um parâmetro de string e apresentará os resultados. Os scripts são executados usando o mecanismo embutido Zabbix Scripting.

Em casos de erros de compilação ou execução, o `zabbix_js` apresentará o erro na saída `stderr` e sairá com código 1.

Uso

```
zabbix_js -s script-file -p input-param [-l log-level] [-t timeout]
zabbix_js -s script-file -i input-file [-l log-level] [-t timeout]
zabbix_js -h
zabbix_js -V
```

`zabbix_js` aceita os seguintes parâmetros de linha de comando:

<code>-s, --script script-file</code>	Specify the file name of the script to execute. If '-' is specified as f
<code>-i, --input input-file</code>	Specify the file name of the input parameter. If '-' is specified as f
<code>-p, --param input-param</code>	Specify the input parameter.
<code>-l, --loglevel log-level</code>	Specify the log level.
<code>-t, --timeout timeout</code>	Specify the timeout in seconds. Valid range: 1-60 seconds (default: 10
<code>-h, --help</code>	Display help information.
<code>-V, --version</code>	Display the version number.

Exemplo:

```
zabbix_js -s script-file.js -p example
```

9 Serviço Web

Visão Geral

O serviço web Zabbix é um processo que é usado para comunicação com serviços web externos. Atualmente, o serviço web Zabbix é usado para gerar e enviar relatórios programados com planos para adicionar funcionalidade no futuro.

O servidor Zabbix se conecta ao serviço web via HTTP(S). O serviço web do Zabbix que o Google Chrome seja instalado no mesmo host; em alguns distribuições o serviço também pode funcionar com o Chromium (consulte [dúvidas frequentes](#)) .

Instalação

O serviço web Zabbix está disponível em pacotes Zabbix pré-compilados disponível para download em [Zabbix site](#). Para compilar [Zabbix web serviço](#) de fontes, especifique a opção de configuração `--enable-webservice`.

Veja também:

- Opções de arquivo de configuração para [zabbix_web_service](#);
- [Configurando relatórios programados](#)

4. Instalação

Por favor, utilize a barra lateral para acessar o conteúdo disponível na seção de Instalação.

1 Obtendo o Zabbix

Visão geral

Há quatro formas de obter o Zabbix:

- Instale-o através dos [pacotes de distribuição](#)
- Baixe o arquivo fonte mais recente e o [compile você mesmo](#)
- Instale a partir dos [contêineres](#)
- Baixe a [aplicação virtual](#)

Para baixar os pacotes de distribuição mais recentes, fontes pré-compilados ou a aplicação virtual, acesse a [página de download do Zabbix](#), onde links diretos para as versões mais recentes são disponibilizados.

Obtendo código fonte do Zabbix

Há várias maneiras de obter o código fonte do Zabbix:

- Você pode [baixar](#) as versões estáveis publicadas no site oficial da Zabbix
- Você pode [baixar](#) compilações noturnas da página oficial do desenvolvedor Zabbix
- Você pode obter a versão de desenvolvimento mais recente no repositório de códigos do Git:
 - A principal localização do repositório completo está em <https://git.zabbix.com/scm/zbx/zabbix.git>
 - As versões Master e demais publicações suportadas também são espelhadas para o Github em <https://github.com/zabbix/zabbix>

Um cliente Git deve ser instalado para clonar o repositório. O pacote do cliente de linha de comando oficial é comumente chamado **git** nas distribuições. Para instalar, por exemplo, no Debian/Ubuntu, execute:

```
sudo apt-get update
sudo apt-get install git
```

Para pegar todo o código do Zabbix, mude para o diretório onde você deseja gravar o código e execute:

```
git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

2 Requisitos

Hardware

Memória

O Zabbix requer memória física e de disco. A quantidade de memória de disco necessária obviamente depende do número de hosts e parâmetros que estão sendo monitorados. Se você planeja manter um longo histórico de parâmetros monitorados, deve pensar em pelo menos alguns gigabytes para ter espaço suficiente para armazenar o histórico no banco de dados. Cada processo do Zabbix daemon requer várias conexões ao servidor de banco de dados. A quantidade de memória alocada para a conexão depende da configuração do mecanismo de banco de dados.

Note:

Quanto mais memória física você tiver, mais rápido o banco de dados (e, portanto, o Zabbix) funcionará.

CPU

O Zabbix e, especialmente, o banco de dados do Zabbix pode requerer recursos de CPU significativos dependendo do número de parâmetros monitorados e o mecanismo de banco de dados selecionada.

Outro hardware

A porta de comunicação serial e o modem serial GSM são necessários para utilizar o suporte de notificação por SMS no Zabbix. O conversor USB para serial também funcionará.

Examples of hardware configuration

The table provides examples of hardware configuration, assuming a **Linux/BSD/Unix** platform.

These are size and hardware configuration examples to start with. Each Zabbix installation is unique. Make sure to benchmark the performance of your Zabbix system in a staging or development environment, so that you can fully understand your requirements before deploying the Zabbix installation to its production environment.

Installation size	Monitored metrics ¹	CPU/vCPU cores	Memory (GiB)	Database	Amazon EC2 ²
Small	1 000	2	8	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.large/m6g.large
Medium	10 000	4	16	MySQL Server, Percona Server, MariaDB Server, PostgreSQL	m6i.xlarge/m6g.xlarge
Large	100 000	16	64	MySQL Server, Percona Server, MariaDB Server, PostgreSQL, Oracle	m6i.4xlarge/m6g.4xlarge

Installation size	Monitored metrics ¹	CPU/vCPU cores	Memory (GiB)	Database	Amazon EC2 ²
Very large	1 000 000	32	96	MySQL Server, Percona Server, MariaDB Server, PostgreSQL, Oracle	m6i.8xlarge/m6g.8xlarge

¹ 1 metric = 1 item + 1 trigger + 1 graph
² Example with Amazon general purpose EC2 instances, using ARM64 or x86_64 architecture, a proper instance type like Compute/Memory/Storage optimised should be selected during Zabbix installation evaluation and testing before installing in its production environment.

Note:

Actual configuration depends on the number of active items and refresh rates very much (see [database size](#) section of this page for details). It is highly recommended to run the database on a separate box for large installations.

Plataformas suportadas

Devido a requisitos de segurança e a natureza crítica do servidor de monitoramento, o UNIX é o único sistema operacional que consegue fornecer consistentemente a performance, tolerância a falhas e resiliência necessárias. O Zabbix opera em versões líderes de mercado.

Os componentes do Zabbix estão disponíveis e foram testados para as seguintes plataformas:

Plataforma	Servidor	Agente	Agente2
Linux	x	x	x
IBM AIX	x	x	-
FreeBSD	x	x	-
NetBSD	x	x	-
OpenBSD	x	x	-
HP-UX	x	x	-
Mac OS X	x	x	-
Solaris	x	x	-
Windows	-	x	x

Note:

O servidor Zabbix/agente Zabbix pode funcionar em outros sistemas operacionais baseados em Unix. O agente Zabbix é suportado em todos os Windows versão desktop e versões de servidor desde o XP.

Attention:

O Zabbix desabilita os core dumps se compilado com criptografia e não inicia se o sistema não permitir a desativação dos core dumps.

Software necessário

O Zabbix é construído em torno de servidores web modernos, principais mecanismos de banco de dados e linguagem de programação PHP.

Third-party external surrounding software

Mandatory requirements are needed always. Optional requirements are needed for the support of the specific function.

Software	Mandatory status	Supported versions	Comments
MySQL/Percona	One of	8.0.X	Required if MySQL (or Percona) is used as Zabbix backend database. InnoDB engine is required. We recommend using the MariaDB Connector/C library for building server/proxy.
MariaDB		10.5.00-10.8.X	InnoDB engine is required. We recommend using the MariaDB Connector/C library for building server/proxy.
Oracle		19c - 21c	Required if Oracle is used as Zabbix backend database.
PostgreSQL		13.0-14.X	Required if PostgreSQL is used as Zabbix backend database.

Software	Mandatory status	Supported versions	Comments
TimescaleDB for PostgreSQL	Optional	2.0.1-2.7	Required if TimescaleDB is used as a PostgreSQL database extension. Make sure to install TimescaleDB Community Edition, which supports compression.
SQLite		3.3.5-3.34.X	SQLite is only supported with Zabbix proxies. Required if SQLite is used as Zabbix proxy database.
smartmontools		7.1 or later	Required for Zabbix agent 2.
who			Required for the user count plugin.
dpkg			Required for the system.sw.packages plugin.
pkgtool			Required for the system.sw.packages plugin.
rpm			Required for the system.sw.packages plugin.
pacman			Required for the system.sw.packages plugin.

Note:

Although Zabbix can work with databases available in the operating systems, for the best experience, we recommend using databases installed from the official database developer repositories.

Frontend

A largura mínima de tela suportada para o frontend Zabbix é 1200px.

Software	Versão	Comentários
Apache	1.3.12 ou posterior	O PHP 8.0 não é suportado.
PHP	7.2.5 ou posterior	
Extensões do PHP		
gd	2.0.28 ou posterior	A extensão PHP GD deve suportar imagens PNG (--with-png-dir), JPEG (--with-jpeg-dir) e FreeType 2 (--with-freetype-dir).
bcmath	2.6.15 ou posterior	php-bcmath (--enable-bcmath)
ctype		php-ctype (--enable-ctype)
libXML		php-xml, se fornecido como um pacote separado pelo distribuidor.
xmlreader		php-xmlreader, se fornecido como um pacote separado pelo distribuidor.
xmlwriter		php-xmlwriter, se fornecido como um pacote separado pelo distribuidor.
session		php-session, se fornecido como um pacote separado pelo distribuidor.
sockets		php-net-socket (--enable-sockets). Necessário para suporte ao script de usuário.
mbstring		php-mbstring (--enable-mbstring)
gettext		php-gettext (--with-gettext). Necessário para que as traduções funcionem.
ldap		php-ldap. Necessário somente se a autenticação por LDAP for utilizada no frontend.
openssl		php-openssl. Necessário somente se a autenticação por SAML for utilizada no frontend.
mysqli		Necessário se o MySQL for utilizado como banco de dados no backend do Zabbix.
oci8		Necessário se o Oracle for utilizado como banco de dados no backend do Zabbix.
pgsql		Necessário se o PostgreSQL for utilizado como banco de dados no backend do Zabbix.

Note:

O Zabbix também pode funcionar em versões anteriores do Apache, MySQL, Oracle e PostgreSQL.

Attention:

Para outras fontes além do padrão DejaVu, a função do PHP `imagerotate` pode ser necessária. Se estiver ausente, essas fontes podem ser renderizadas incorretamente quando um gráfico é exibido. Esta função só está disponível se o PHP for compilado com GD empacotado, o que não é o caso no Debian e outros distribuições.

Navegador da web no lado do cliente

Os Cookies e o JavaScript devem estar habilitados.

As últimas versões estáveis do Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari e Opera são suportadas.

Warning:

A política de mesma origem para os IFrames é implementada, o que significa que o Zabbix não pode ser inserido em frames em domínio diferente.

Ainda assim, as páginas colocadas em um frame do Zabbix terão acesso ao Zabbix frontend (através de JavaScript) se a página que é inserida no frame e o frontend do Zabbix estiverem no mesmo domínio. Uma página como `http://secure-zabbix.com/cms/page.html`, se inserida nos dashboards em `http://secure-zabbix.com/zabbix/`, terá o total acesso JS ao Zabbix.

Servidor

Requisitos obrigatórios são necessários sempre. Os requisitos opcionais são necessários para o suporte da função específica.

Requisito	Status	Descrição
libpcre	Obrigatório	A biblioteca PCRE é necessária para suporte ao Perl Compatible Regular Expression (PCRE). A nomenclatura pode diferir dependendo da distribuição GNU/Linux, por exemplo 'libpcre3' ou 'libpcre1'. PCRE v8.x e PCRE2 v10.x (do Zabbix 6.0.0) são suportados.
libevent		Necessário para suporte de métrica em massa e monitoramento de IPMI. Versão 1.4 ou superior. Observe que para o proxy Zabbix este requisito é opcional; é necessário para suporte de monitoramento IPMI.
libpthread		Required for mutex and read-write lock support.
zlib	Optional	Necessário para o suporte a compressão.
OpenIPMI		Necessário para suporte IPMI.
libssh2 or libssh		Necessário para SSH checks . Versão 1.0 ou superior (libssh2); 0.6.0 ou superior (libssh). libssh é suportado desde o Zabbix 4.4.6.
fping		Necessário para ICMP ping items .
libcurl		Necessário para monitoramento web, monitoramento VMware, autenticação SMTP, <code>web.page.* items</code> do agente Zabbix, itens do agente HTTP e Elasticsearch (se utilizado). Recomendado versão 7.28.0 ou superior. Requisitos da versão do Libcurl: - Autenticação SMTP: versão 7.20.0 ou superior - Elasticsearch: versão 7.28.0 ou superior
libxml2		Necessário para monitoramento VMware e pré-processamento XML XPath.
net-snmp		Necessário para suporte SNMP. Versão 5.3.0 ou superior.
GnuTLS, OpenSSL ou LibreSSL		Necessário ao utilizar encryption .

Agente

Requisito	Status	Descrição
libpcre	Obrigatório	A biblioteca PCRE é obrigatória para suporte ao Perl Compatible Regular Expression (PCRE). A nomenclatura pode diferir dependendo da distribuição GNU/Linux, por exemplo 'libpcre3' ou 'libpcre1'. PCRE v8.xe PCRE2 v10.x (do Zabbix 6.0.0) são suportados.
GnuTLS, OpenSSL ou LibreSSL	Opcional	Necessário ao utilizar encryption . Em sistemas Microsoft Windows é necessário OpenSSL 1.1.1 ou posterior.

Note:

A partir da versão 5.0.3, o agente Zabbix não funcionará em plataformas AIX inferior as versões 6.1 TL07 / AIX 7.1 TL01.

Agent 2

Requisito	Status	Descrição
libpcre	Obrigatório	A biblioteca PCRE é obrigatória para suporte ao Perl Compatible Regular Expression (PCRE). A nomenclatura pode diferir dependendo da distribuição GNU/Linux, por exemplo 'libpcre3' ou 'libpcre1'. PCRE v8.xe PCRE2 v10.x (do Zabbix 6.0.0) são suportados.
OpenSSL	Opcional	Necessário ao usar criptografia. OpenSSL 1.0.1 ou posterior é necessário em plataformas UNIX. A biblioteca OpenSSL deve ter o suporte PSK ativado. O LibreSSL não é suportado. Nos sistemas Microsoft Windows, o OpenSSL 1.1.1 ou posterior é necessário.

Java gateway

Se você obteve o Zabbix do repositório de origem ou de um arquivo, então as dependências necessárias já estão incluídas na árvore de origem.

Se você obteve o Zabbix do pacote de sua distribuição, então as dependências necessárias já são fornecidas pelo sistema de empacotamento.

Em ambos os casos acima, o software está pronto para ser usado e não há downloads são necessários.

Se, no entanto, você deseja fornecer suas versões dessas dependências (por exemplo, se você estiver preparando um pacote para alguns distribuição), abaixo está a lista de versões de biblioteca que o gateway Java é conhecido por trabalhar. O Zabbix pode funcionar com outras versões destas bibliotecas também.

A tabela a seguir lista os arquivos JAR que estão atualmente empacotados com Java gateway no código original:

Biblioteca	Licença	Site	Comentários
logback-core-1.2.3.jar	EPL 1.0, LGPL 2.1	http://logback.qos.ch/	Testado com 0.9.27, 1.0.13, 1.1.1 e 1.2.3.
logback-classic-1.2.3.jar	EPL 1.0, LGPL 2.1	http://logback.qos.ch/	Testado com 0.9.27, 1.0.13, 1.1.1 e 1.2.3.
slf4j-api-1.7.30.jar	Licença MIT	http://www.slf4j.org/	Testado com 1.6.1, 1.6.6, 1.7.6 e 1.7.30.
android-json-4.3_r3.1.jar	Licença Apache 2.0	https://android.googlesource.com/platform/libcore/+/_master/json	Testado com 2.3.3_r1.1 e 4.3_r3.1. Verifique src/zabbix_java/lib/README para instruções de criação de arquivo JAR.

O Java gateway pode ser construído usando Oracle Java ou código aberto OpenJDK (versão 1.6 ou mais recente). Os pacotes fornecidos pelo Zabbix são compilados usando o OpenJDK. A tabela abaixo fornece informações sobre as versões do OpenJDK usadas para construir pacotes Zabbix por distribuição:

Distribuição	Versão do OpenJDK
RHEL/CentOS 8	1.8.0
RHEL/CentOS 7	1.8.0
SLES 15	11.0.4
SLES 12	1.8.0
Debian 10	11.0.8
Ubuntu 20.04	11.0.8
Ubuntu 18.04	11.0.8

Default port numbers

The following list of open ports per component is applicable for default configuration:

Zabbix component	Port number	Protocol	Type of connection
Zabbix agent	10050	TCP	on demand
Zabbix agent 2	10050	TCP	on demand
Zabbix server	10051	TCP	on demand
Zabbix proxy	10051	TCP	on demand
Zabbix Java gateway	10052	TCP	on demand
Zabbix web service	10053	TCP	on demand
Zabbix frontend	80	HTTP	on demand
	443	HTTPS	on demand
Zabbix trapper	10051	TCP	on demand

Note:

The port numbers should be open in firewall to enable Zabbix communications. Outgoing TCP connections usually do not require explicit firewall settings.

Tamanho do banco de dados

Os dados de configuração do Zabbix requerem uma quantidade fixa de espaço em disco e não crescem muito.

O tamanho do banco de dados Zabbix depende principalmente dessas variáveis, que definem a quantidade de dados históricos armazenados:

- Números de valores processados por segundo

Este é o número médio de novos valores que o servidor Zabbix recebe a cada segundo. Por exemplo, se tivermos 3.000 itens para monitoramento com uma taxa de atualização de 60 segundos, o número de valores por segundo será calculado como $3.000/60 = 50$.

Isso significa que 50 novos valores são adicionados ao banco de dados Zabbix a cada segundo.

- Configuração do Housekeeper para o histórico

Zabbix mantém valores por um período fixo de tempo, normalmente várias semanas ou meses. Cada novo valor requer uma certa quantidade de espaço em disco para dados e índice.

Então, se quisermos manter 30 dias de histórico e recebermos 50 valores por segundo, o número total de valores será em torno de $(30 \times 24 \times 3600) \times 50 = 129.600.000$, ou cerca de 130M de valores.

Dependendo do mecanismo de banco de dados usado, tipo de valores recebidos (reais, inteiros, strings, arquivos de log, etc), o espaço em disco para manter um único valor pode variar de 40 bytes a centenas de bytes. Normalmente é cerca de 90 bytes por valor para itens numéricos². No nosso caso, significa que 130M de valores exigirão $130M \times 90 \text{ bytes} = 10.9GB$ de espaço em disco.

Note:

O tamanho dos itens do tipo texto/log é impossível de ser previsto com exatidão, mas você esperar em torno de 500 bytes por valor.

- Configuração do Housekeeper para a estatística

O Zabbix mantém um conjunto de valores máximo/min/médio/contagem de 1 hora para cada item na tabela **trends**. Os dados são usados para tendências e gráficos de longos períodos. O período de uma hora não pode ser personalizado.

O banco de dados do Zabbix, dependendo do tipo do banco de dados, requer cerca de 90 bytes para cada total. Suponha que gostaríamos de manter os dados de estatística por 5 anos. Os valores para 3.000 itens exigirão $3.000 \times 24 \times 365 \times 90 = 2,2 \text{ GB}$ por ano ou **11 GB** por 5 anos.

- Configuração do Housekeeper para os eventos

Cada evento Zabbix requer aproximadamente 250 bytes de espaço em disco¹. É difícil estimar o número de eventos gerados diariamente pelo Zabbix. Na pior das hipóteses, podemos assumir que o Zabbix gera um evento por segundo.

Para cada evento de recuperação, é criado um registro event_recovery. Normalmente a maioria dos eventos serão recuperados para que possamos assumir um registro de event_recovery por evento. Isso significa 80 bytes adicionais por evento.

Opcionalmente, os eventos podem ter tags, cada registro de tag requer aproximadamente 100 bytes de espaço em disco¹. O número de tags por evento (#tags) depende da configuração. Portanto, cada um precisará de mais #tags * 100 bytes de espaço em disco.

Isso significa que se quisermos manter 3 anos de eventos, isso exigiria $3 \times 365 \times 24 \times 3600 \times (250 + 80 + \#tags \times 100) = \sim 30GB + \#tags \times 100B$ disco espaço².

Note:

¹ Mais quando tiver nomes de eventos não ASCII, tags e valores.

² As aproximações de tamanho são baseadas no MySQL e podem ser diferentes para outros bancos de dados.

A tabela contém fórmulas que podem ser usadas para calcular o espaço em disco necessário para o sistema Zabbix:

Parâmetro	Fórmula para o espaço em disco necessário (em bytes)
Configuração do Zabbix	Tamanho fixo. Normalmente 10MB ou menos.
Histórico	$dias \times (itens / \text{intervalo de atualização}) \times 24 \times 3600 \times \text{bytes}$ itens : número de itens dias : número de dias para manter o histórico intervalo de atualização : intervalo de atualização médio para os itens bytes : número de bytes necessário para manter um valor, depende do mecanismo de banco de dados, normalmente ~90 bytes.
Estatísticas	$dias \times (itens / 3600) \times 24 \times 3600 \times \text{bytes}$ itens : número de itens dias : número de dias para manter o histórico bytes : número de bytes necessário para manter um valor estatístico, depende do mecanismo de banco de dados, normalmente ~90 bytes.
Eventos	$dias \times \text{eventos} \times 24 \times 3600 \times \text{bytes}$ eventos : número de eventos por segundo. Um (1) evento por segundo no pior cenário. dias : número de dias para manter o histórico bytes : número de bytes necessários para manter um valor estatístico, depende do mecanismo de banco de dados, normalmente ~330 + número médio de tags por evento * 100 bytes.

Assim, o espaço total em disco necessário pode ser calculado como:

Configuração + Histórico + Estatísticas + Eventos

O espaço em disco NÃO será usado imediatamente após a instalação do Zabbix. O tamanho do banco de dados aumentará e depois parará de crescer em algum ponto, o que depende das configurações do Housekeeper.

Sincronização de tempo

É muito importante ter a hora exata do sistema no servidor no qual o Zabbix está em execução. O [ntpd](#) é o daemon mais popular que sincroniza a hora do host com a hora de outras máquinas. Isso é fortemente recomendado para manter a hora do sistema sincronizada em todos os sistemas nos quais os componentes do Zabbix estão sendo executados.

1 Plugins

Por favor utilize a barra lateral para acessar o conteúdo na seção do Plugin.

1 Dependências do plugin PostgreSQL

Visão geral

As bibliotecas necessárias para o plugin PostgreSQL carregável estão listada nesta página.

Bibliotecas Go

Requirement	Status obri- gatório	Versão mínima	Descrição
git.zabbix.com/ap/plugin-support	Sim	1.X.X	Biblioteca de suporte própria do Zabbix. Principalmente para plugins.
github.com/jackc/pgx/v4		4.17.2	PostgreSQL driver.
github.com/omeid/go-yarn		0.0.1	Armazenamento incorporável de key-string mapeado em sistemas de arquivos.
github.com/jackc/chunkin	Indireto ¹	2.0.1	
github.com/jackc/pgconn		1.13.0	
github.com/jackc/pgio		1.0.0	
github.com/jackc/pgpassfile		1.0.0	
github.com/jackc/pgproto3		2.3.1	
github.com/jackc/pgservicefile		0.0.0	
github.com/jackc/pgtype		1.12.0	
github.com/jackc/puddle		1.3.0	
github.com/Microsoft/go-winio		0.6.0	Pacote necessário para plugin PostgreSQL no Windows.
golang.org/x/crypto		0.0.0	
golang.org/x/sys		0.0.0	
golang.org/x/text		0.3.7	

¹ "Indireto" significa que é usado em umas das bibliotecas que o agente utiliza. Isso é necessário, já que o Zabbix utiliza a biblioteca que usa o pacote.

2 Dependências do plugin MongoDB

Visão geral

As bibliotecas necessárias para o plugin carregável MongoDB, estão listadas nesta página.

Bibliotecas Go

Requerimento	Status obri- gatório	Versão mínima	Descrição
git.zabbix.com/ap/plugin-support	Sim	1.X.X	Biblioteca de suporte própria do Zabbix. Principalmente para plugins.
go.mongodb.org/mongo-driver		1.7.6	Bloqueios nomeados read/write, sincronização de acesso.
github.com/go-stack/stack	Indirect ¹	1.8.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/golang/snappy		0.0.1	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/klauspost/compress		1.13.6	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/Microsoft/go-winio		0.6.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/pkg/errors		0.9.1	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/xdg-go/pbkdf2		1.0.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/xdg-go/scram		1.0.2	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/xdg-go/stringprep		1.0.2	Pacote necessário para o plugin MongoDB é mongo-driver lib.
github.com/youmark/pkcs8		0.0.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.

Requerimento	Status obri- gatório	Versão mínima	Descrição
golang.org/x/crypto		0.0.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
golang.org/x/sync		0.0.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
golang.org/x/sys		0.0.0	Pacote necessário para o plugin MongoDB é mongo-driver lib.
golang.org/x/text		0.3.7	Pacote necessário para o plugin MongoDB é mongo-driver lib.

¹ "Indirect" significa que é usado em uma das bibliotecas que o agente utiliza. É necessário, pois o Zabbix utiliza a biblioteca que utiliza os pacotes.

2 Melhores práticas para configuração segura do Zabbix

Visão geral

Esta seção contém boas práticas que devem ser observadas de modo a configurar o Zabbix de uma forma segura.

As práticas contidas aqui não são necessárias para o funcionamento do Zabbix. Elas são recomendadas para uma melhor segurança do sistema.

Controle de acesso

Princípio do menor privilégio

O princípio do menor privilégio deve ser usado todo o tempo no Zabbix. Este princípio implica que contas de usuário (no Zabbix Frontend) ou usuários de processo (para Zabbix Server/Proxy ou Agent) tenham apenas aqueles privilégios essenciais para executar as funções pretendidas. Em outras palavras, contas de usuário devem fornecer o mínimo de privilégios possível, durante todo o tempo.

Attention:

Fornecer permissões extras ao usuário 'zabbix' permitirá que este acesse os arquivos de configuração e execute operações que podem comprometer a segurança geral da infraestrutura.

Quando implementando o princípio de mínimo privilégio para contas de usuário, os **tipos de usuário do frontend** do Zabbix devem ser levados em conta. É importante entender que enquanto um usuário do tipo "Admin" tem menos privilégios do um usuário do tipo "Super Admin", ele tem permissões administrativas que o permitem gerenciar configurações e executar scripts customizados.

Note:

Algumas informações estão disponíveis até mesmo para usuários sem privilégio. Por exemplo, conquanto Administração → Scripts esteja disponível apenas para usuários Super Admins, os próprios scripts estão disponíveis para recuperação através do uso da API do Zabbix. Limitação nas permissões dos scripts e a não adição de informações sensíveis (como credenciais de acesso, etc) devem ser consideradas para evitar a exposição de informações sensíveis existentes nos scripts globais.

Usuário seguro para Zabbix Agent

Na configuração padrão, os processos do Zabbix Server e Zabbix Agent compartilham um usuário 'zabbix'. Se você desejar certificar-se de que o Agent não tenha acesso a detalhes sensíveis na configuração do Server (p.e. informações de login do banco de dados), o Agent deve ser executado com um usuário diferente:

1. Crie um usuário seguro
2. Especifique este usuário no **arquivo de configuração** (parâmetro 'User') do Agent
3. Reinicie o Agent com privilégios de administrador. Estes privilégios serão substituídos pelos privilégios do usuário especificado.

Codificação UTF-8

O UTF-8 é o único formato de codificação suportado pelo Zabbix. É conhecido por operar sem quaisquer falhas de segurança. Usuários devem estar cientes de que há problemas de segurança conhecidos quando usando algum dos outros formatos.

Caminhos para instaladores para Windows

Ao utilizar instaladores para Windows, é recomendável utilizar os caminhos padrão fornecidos pelo instalador. Usar caminhos personalizados sem as permissões adequadas pode comprometer a segurança da instalação.

Avisos de segurança Zabbix e banco de dados CVE

Consulte [Zabbix Security Advisories and CVE database](#).

Configurando SSL para Zabbix Frontend

No RHEL/Centos, instale o pacote mod_ssl:

```
yum install mod_ssl
```

Crie diretório para as chaves SSL:

```
mkdir -p /etc/httpd/ssl/private
chmod 700 /etc/httpd/ssl/private
```

Crie o certificado SSL:

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/httpd/ssl/private/apache-selfsigned.key -
```

Preencha cada informação com os dados apropriados. A linha mais importante é a que solicita o Nome Comum (Common Name). Aqui você precisa informar o nome de domínio que deseja que seja associado ao seu servidor. Você pode informar o endereço de IP público caso não tenha um nome de domínio. Neste artigo nós faremos uso do nome example.com.

```
Country Name (2 letter code) [XX]:
State or Province Name (full name) []:
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:example.com
Email Address []:
```

Edite as configurações de SSL do Apache:

```
/etc/httpd/conf.d/ssl.conf
```

```
DocumentRoot "/usr/share/zabbix"
ServerName example.com:443
SSLCertificateFile /etc/httpd/ssl/apache-selfsigned.crt
SSLCertificateKeyFile /etc/httpd/ssl/private/apache-selfsigned.key
```

Reinicie o serviço do Apache para aplicar as alterações:

```
systemctl restart httpd.service
```

Fortalecimento do servidor Web

Habilitando o Zabbix no diretório raiz da URL

Adicione um virtual host na configuração do Apache e configure um redirecionamento permanente do diretório raiz (DocumentRoot) para a URL com SSL do Zabbix. Não esqueça de substituir example.com pelo nome real do servidor.

```
/etc/httpd/conf/httpd.conf
```

#Linhas adicionais

```
<VirtualHost *:*>
    ServerName example.com
    Redirect permanent / https://example.com
</VirtualHost>
```

Reinicie o serviço do Apache para aplicar as alterações:

```
systemctl restart httpd.service && systemctl status httpd.service
```

Habilitando HTTP Strict Transport Security (HSTS) no servidor web

Para proteger o Zabbix Frontend contra ataques de rebaixamento de protocolo, nós recomendamos habilitar a política de [HSTS](#) no servidor web.

Por exemplo, para habilitar a política HSTS para seu Zabbix Frontend na configuração do Apache:

```
/etc/httpd/conf/httpd.conf
```

adicione a seguinte diretiva à configuração do seu virtual host:

```
<VirtualHost *:443>
    Header set Strict-Transport-Security "max-age=31536000"
```



```
</VirtualHost>
```

Reinicie o serviço do Apache para aplicar as alterações:

```
systemctl restart httpd.service
```

Habilitando a Política de Segurança do Conteúdo no servidor web

Para proteger o frontend Zabbix contra Cross Site Scripting (XSS), injeção de dados, e ataques similares, recomendamos habilitar a Política de Segurança do Conteúdo no servidor web. Para isso, configure o servidor web para retornar [HTTP header](#).

Attention:

A seguinte configuração do cabeçalho CSP é apenas para instalação do frontend Zabbix padrão e para casos em que todos os conteúdos são produzidos a partir do domínio do site (excluindo subdomínios). Uma configuração diferente do cabeçalho CSP pode ser necessária se você estiver, por exemplo, configurando o widget **URL** para exibir o conteúdo dos subdomínios do site ou a partir de domínios externos, alternando de OpenStreetMap para um outro mecanismo de mapa, ou adicionando CSS externo ou widgets.

Para habilitar o CSP para seu frontend Zabbix na configuração Apache, siga esses passos:

1. Localize seu arquivo de configuração do host virtual:

- /etc/httpd/conf/httpd.conf em RHEL-based systems
- /etc/apache2/sites-available/000-default.conf em Debian/Ubuntu

2. Adicione o seguinte diretivo para o seu arquivo de configuração do host virtual:

```
<VirtualHost *:*>
    Header set Content-Security-Policy: "default-src 'self' *.openstreetmap.org; script-src 'self' 'unsafe-inline'
</VirtualHost>
```

3. Reinicie o serviço Apache para aplicar todas as alterações:

```
#### On RHEL-based systems:
systemctl restart httpd.service
```

```
#### On Debian/Ubuntu
systemctl restart apache2.service
```

Desabilitando exposição de informação do servidor web

É recomendado desabilitar todas as assinaturas do web server como parte do processo de garantia da segurança. O Web Server expõe sua assinatura de software por padrão:

▼ **Response Headers** [view source](#)

```
Cache-Control: no-store, no-cache, must-revalidate
Connection: Keep-Alive
Content-Encoding: gzip
Content-Length: 1160
Content-Type: text/html; charset=UTF-8
Keep-Alive: timeout=5, max=100
Pragma: no-cache
Server: Apache/2.4.18 (Ubuntu)
```

A assinatura pode ser desabilitada através da adição de duas linhas no arquivo de configuração do Apache, por exemplo::

```
ServerSignature Off
ServerTokens Prod
```

A assinatura do PHP (X-Powered-By HTTP header) pode ser desabilitada pela alteração do arquivo de configuração php.ini (neste caso a assinatura é desabilitada por padrão):

```
expose_php = Off
```

Um reinício do Web Server é necessário para que as alterações no arquivo de configuração sejam aplicadas.

Um nível adicional de segurança pode ser alcançado usando `mod_security` (pacote `libapache2-mod-security2`) com Apache. O `mod_security` permite remover toda a assinatura do server em vez de apenas remover a versão. A assinatura pode ser alterada para qualquer valor pela alteração de `"SecServerSignature"` para um valor desejado após a instalação do `mod_security`.

Por favor, consulte a documentação do seu Web Server para encontrar auxílio em como remover/alterar as assinaturas de software.

Desabilitando as páginas de erro padrão do Web Server

É recomendado desabilitar as páginas de erro padrão para evitar exposição de informação. O Web Server usa as páginas de erro embutidas por padrão:

Not Found

The requested URL `/custom-text` was not found on this server.

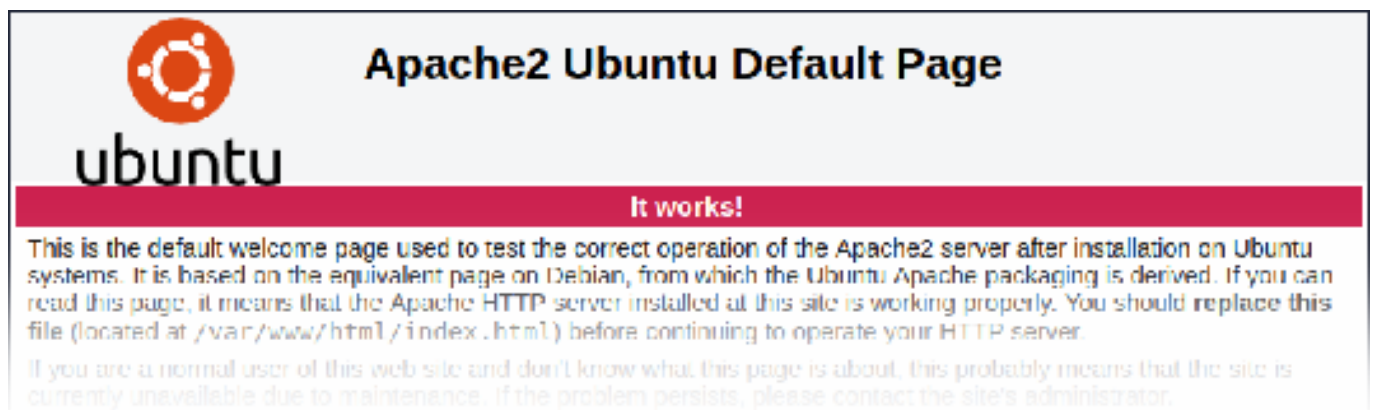
Apache/2.4.18 (Ubuntu) Server at localhost Port 80

Páginas de erro padrão devem ser substituídas/removidas como parte do processo de aprimoramento da segurança. A diretiva `"ErrorDocument"` pode ser usada para definir uma página/texto de erro customizado para o Apache Web Server (usado como exemplo).

Por favor, consulte a documentação do seu Web Server para encontrar auxílio em como substituir/remover as páginas de erro padrão.

Removendo a página de teste do Web Server

É recomendado remover a página de teste do Web Server para evitar a exposição de informações. Por padrão, o diretório raiz do Web Server contém uma página de teste chamada `index.html` (usando Apache2 no Ubuntu como exemplo):



A página de teste deve ser removida ou tornada indisponível como parte do processo de aprimoramento da segurança do Web Server.

Configurações do Zabbix

Por padrão, o Zabbix possui a opção `X-Frame-Options` HTTP response header configurada como `SAMEORIGIN`, significando que o conteúdo só pode ser carregado em um frame que tenha a mesma origem da página em si.

Os elementos do Zabbix Frontend que buscam conteúdo de URLs externas (especificamente, o **widget de URL para dashboard**) apresentam o conteúdo encontrado em uma área isolada (sandbox) com todas as restrições habilitadas.

Estas configurações aprimoram a segurança do Zabbix Frontend e proveem proteção contra ataques do tipo XSS e clickjacking. Super Admins podem **modificar** os parâmetros `iframe sandboxing` e `X-Frame-Options` HTTP response header conforme necessário. Por favor, pese cuidadosamente os riscos e benefícios antes de alterar as configurações padrão. Desativar completamente a função de sandboxing ou `X-Frame-Options` não é recomendado.

Zabbix Windows Agent com OpenSSL

O Zabbix Windows Agent compilado com OpenSSL tentará encontrar o arquivo de configuração SSL em C:\openssl-64bit. O diretório "openssl-64bit" na partição C: pode ser criado por usuários sem privilégio elevado.

Assim, para aprimoramento da segurança, é necessária a criação deste diretório manualmente e então a retirada do acesso de escrita para os usuários que não sejam administradores.

Por favor, note que os nomes do diretório serão diferentes nas versões 32-bit e 64-bit do Windows.

Criptografia

Escondendo o arquivo com lista de senhas comuns

Para aumentar a complexidade dos ataques de força bruta a senhas, sugere-se limitar o acesso ao arquivo `ui/data/top_passwords.txt`, modificando a configuração do servidor web. Este arquivo contém uma lista de senhas mais comuns e senhas específicas ao contexto, sendo utilizado para evitar que usuários definam tais senhas caso o parâmetro `Avoid easy-to-guess passwords` esteja habilitado na **password policy**.

Por exemplo, o acesso ao arquivo NGINX pode ser limitado usando a diretiva `location`:

```
location = /data/top_passwords.txt {
    deny all;
    return 404;
}
```

No Apache - usando o arquivo `.htaccess`:

```
<Files "top_passwords.txt">
    Order Allow,Deny
    Deny from all
</Files>
```

3 Instalação a partir do código-fonte

É possível acessar a versão mais recente do Zabbix compilando-a a partir do código-fonte.

A seguir, fornecemos um tutorial com os passos para instalação do Zabbix a partir do código-fonte.

1 Instalando daemons do Zabbix

1 Download do arquivo fonte

Vá até a página de [download do Zabbix](#) e baixe o arquivo fonte. Uma vez baixado, extraia os fontes, executando:

```
$ tar -zxvf zabbix-6.0.0.tar.gz
```

Note:

Informe a versão correta do Zabbix no comando. Ela deve corresponder ao nome do arquivo baixado.

2 Criar conta de usuário

Para todos os processos de daemon do Zabbix, um usuário sem privilégios é requerido. Se um daemon Zabbix é iniciado a partir de uma conta de usuário sem privilégios, ele será executado como tal usuário.

No entanto, se um daemon é iniciado a partir de uma conta 'root', ele mudará para uma conta de usuário 'zabbix', que deve existir. Para criar esta conta de usuário (em seu próprio grupo, "zabbix"),

em sistemas baseados em RedHat, execute:

```
groupadd --system zabbix
useradd --system -g zabbix -d /usr/lib/zabbix -s /sbin/nologin -c "Zabbix Monitoring System" zabbix
```

em sistemas baseados em Debian, execute:

```
addgroup --system --quiet zabbix
adduser --quiet --system --disabled-login --ingroup zabbix --home /var/lib/zabbix --no-create-home zabbix
```

Attention:

Os processos Zabbix não precisam de um diretório home, sendo este o motivo de não recomendarmos a sua criação. No entanto, se você estiver usando alguma funcionalidade que demande tal diretório (p.e. armazenar credenciais MySQL em `$HOME/.my.cnf`), você está livre para criá-lo usando os seguintes comandos:

```
Em sistemas baseados em RedHat, execute:
mkdir -m u=rwx,g=rwx,o=-p /usr/lib/zabbix
chown zabbix:zabbix /usr/lib/zabbix
Em sistemas baseados em Debian, execute:
mkdir -m u=rwx,g=rwx,o=-p /var/lib/zabbix
chown zabbix:zabbix /var/lib/zabbix
```

Não é necessária uma conta de usuário separada para a instalação do Zabbix Frontend.

Se o Zabbix **Server** e **Agent** são executados na mesma máquina é recomendado usar um usuário diferente para executar o server daquele usado para o agent. Caso contrário, se ambos são executados com o mesmo usuário, o agent pode acessar os arquivos de configuração do server e assim qualquer usuário do Zabbix com perfil de Admin pode facilmente, por exemplo, descobrir a senha do banco de dados.

Attention:

Executar o Zabbix como root, bin, ou qualquer outra conta com permissões especiais é um risco à segurança.

3 Criar banco de dados Zabbix

Para os daemons do Zabbix **Server** e **Proxy**, bem como para o Zabbix Frontend, um banco de dados é necessário. No entanto não é necessário para executar o Zabbix **Agent**.

Scripts SQL são disponibilizados para criação do schema de banco de dados e carga do conjunto de dados iniciais. O banco de dados do Zabbix Proxy necessita apenas do schema enquanto que o banco de dados do Zabbix Server requer também o conjunto de dados junto ao schema.

Tendo criado um banco de dados para o Zabbix, prossiga para os próximos passos de compilação do Zabbix.

4 Configurar os fontes

Ao configurar os fontes para o Zabbix Server ou Proxy, você deve especificar o tipo de banco de dados a ser usado. Apenas um tipo de banco de dados pode ser compilado com um processo Server ou Proxy por vez.

Para ver todas as opções de configuração suportadas, dentro do diretório de fonte do Zabbix extraído, execute:

```
./configure --help
```

Para configurar os fontes para o Zabbix Server e Agent, você pode executar algo como o seguinte:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --with-
```

Para configurar os fontes para o Zabbix Server (com PostgreSQL, etc.), você pode usar:

```
./configure --enable-server --with-postgresql --with-net-snmp
```

Para configurar os fontes para Zabbix Proxy (com SQLite, etc.), você pode executar:

```
./configure --prefix=/usr --enable-proxy --with-net-snmp --with-sqlite3 --with-ssh2
```

Para configurar os fontes para o Zabbix Agent, você pode utilizar:

```
./configure --enable-agent
```

ou, para Zabbix Agent 2:

```
./configure --enable-agent2
```

Note:

Um ambiente Go configurado com uma [versão atualmente suportada](#) é necessário para compilar o Zabbix Agent 2. Consulte [golang.org](#) para instruções de instalação.

Notas sobre as opções de compilação:

- Os utilitários de linha de comando `zabbix_get` e `zabbix_sender` são compilados se a opção `--enable-agent` é usada.

- As opções de configuração `--with-libcurl` e `--with-libxml2` são necessárias para monitoramento de máquinas virtuais; `--with-libcurl` também é necessário para autenticação SMTP e `items web.page.*` do Zabbix Agent. Note que o cURL na versão 7.20.0 ou maior é **necessário** com a opção de configuração `--with-libcurl`.
- O Zabbix sempre compila com a biblioteca PCRE (desde a versão 3.4.0); sua instalação não é opcional. `--with-libpcre=[DIR]` apenas permite o direcionamento para um diretório de instalação específico, em vez de buscar entre um número de lugares comuns pelos arquivos da biblioteca `libpcre`.
- Você pode usar o marcador `--enable-static` para vincular bibliotecas de forma estática. Se você planeja distribuir binários compilados entre diferentes servidores, você deve usar este marcador para que tais binários funcionem sem as bibliotecas exigidas. Note que `--enable-static` não funciona em [Solaris](#).
- O uso da opção `--enable-static` não é recomendado quanto compilando o server. Para compilar o server estaticamente, você deve ter uma versão estática de cada biblioteca externa necessária. Não há uma verificação rigorosa quanto a isso no script de configuração.
- Adicione um caminho opcional para o arquivo de configuração do MySQL `--with-mysql=``<path_to_the_file>/mysql_config` para selecionar a biblioteca de cliente MySQL desejada quando houver necessidade de utilizar uma que não esteja localizada no local padrão. Isto é útil quando há várias versões de MySQL instaladas ou um MariaDB instalado junto ao MySQL no mesmo sistema.
- Use o marcador `--with-oracle` para especificar a localização da API OCI.

Attention:

Se `.configure` falhar devido bibliotecas ausentes ou alguma outra circunstância, por favor verifique o `config.log` para mais detalhes quanto ao erro. Por exemplo, se `libssl` está ausente, a mensagem de erro imediata pode ser confusa:

```
checking for main in -lmysqlclient... no
configure: error: Not found mysqlclient library
Enquanto que o config.log possui uma descrição mais detalhada:
/usr/bin/ld: cannot find -lssl
/usr/bin/ld: cannot find -lcrypto
```

Veja também:

- [Compilando Zabbix com suporte à criptografia](#)
- [Problemas conhecidos](#) na compilação do Zabbix Agent em HP-UX

5 Make e install para tudo

Note:

Se instalando através do [repositório Git do Zabbix](#), é necessário primeiro executar:

```
$ make dbschema
```

`make install`

Esta etapa deve ser executada usando um usuário com permissões suficientes (comumente 'root', ou utilizando elevação de usuário com `sudo`).

A execução do comando `make install` instalará, por padrão, os binários dos daemons (`zabbix_server`, `zabbix_agentd`, `zabbix_proxy`) em `/usr/local/sbin` e os binários de client (`zabbix_get`, `zabbix_sender`) em `/usr/local/bin`.

Note:

Para especificar um local diferente de `/usr/local`, use a chave `--prefix` na etapa anterior de configuração dos fontes, por exemplo `--prefix=/home/zabbix`. Neste caso os binários dos daemons seriam instalados dentro de `<prefix>/sbin`, enquanto que os utilitários em `<prefix>/bin`. As páginas de manual seriam instaladas em `<prefix>/share`.

6 Revise e edite os arquivos de configuração

- edite o arquivo de configuração do Zabbix Agent **`/usr/local/etc/zabbix_agentd.conf`**

Você precisa configurar este arquivo para cada máquina com `zabbix_agentd` instalado.

Você deve especificar o **endereço IP** do servidor Zabbix no arquivo. Conexões de outros endereços serão negadas.

- edite o arquivo de configuração do Zabbix Server **`/usr/local/etc/zabbix_server.conf`**

Você deve especificar o nome do banco de dados, usuário e senha (se usando alguma).

O restante dos parâmetros o atenderão com seus valores padrão se você tiver um ambiente pequeno (até 10 máquinas monitoradas). Todavia, você deve alterar os parâmetros padrão se você pretende maximizar a performance do Zabbix Server (ou Proxy). Consulte a seção [ajustes de performance](#) para mais detalhes.

- se você tem instalado um Zabbix Proxy, edite o arquivo de configuração **`/usr/local/etc/zabbix_proxy.conf`**

Você deve especificar o endereço IP e o nome do servidor do Zabbix Proxy (que deve ser conhecido pelo Zabbix Server), bem como o nome do banco de dados, usuário e senha (se usando alguma).

Note:

Com SQLite deve ser especificado o caminho completo para o arquivo de banco de dados; Usuário e senha do banco de dados não são exigidos.

7 Inicie os daemons

Execute o `zabbix_server` no lado do servidor:

```
shell> zabbix_server
```

Note:

Certifique-se de que seu sistema permita a alocação de 36MB (ou um pouco mais) de memória compartilhada, caso contrário o Server pode não iniciar e você verá a mensagem "Impossível alocar memória compartilhada para <tipo de cache>." (se log em inglês: "Cannot allocate shared memory for <type of cache>.") no arquivo de log. Isto pode ocorrer com FreeBSD, Solaris 8.

Verifique a seção "**Veja também**" ao final desta página para descobrir como configurar a memória compartilhada.

Execute `zabbix_agentd` em todas as máquinas monitoradas:

```
shell> zabbix_agentd
```

Note:

Certifique-se de que seu sistema permita a alocação de 2MB de memória compartilhada, caso contrário o Agente pode não iniciar e você verá a mensagem "Impossível alocar memória compartilhada para o coletor" (se log em inglês: "Cannot allocate shared memory for collector.") no arquivo de log. Isto pode ocorrer em Solaris 8.

Se você tem instalado o Zabbix Proxy, execute `zabbix_proxy`.

```
shell> zabbix_proxy
```

2 Instalação da interface web do Zabbix

Copiando arquivos PHP

O frontend do Zabbix é escrito em PHP, assim, para executá-lo, é necessário um webserver com suporte ao PHP. A instalação é feita com a simples cópia dos arquivos PHP do diretório `ui` para o diretório de documentos HTML do webserver.

Locais comuns para o diretório de documentos HTML no Apache Web Server incluem:

- `/usr/local/apache2/htdocs` (diretório padrão quando instalando o Apache a partir do fonte)
- `/srv/www/htdocs` (OpenSUSE, SLES)
- `/var/www/html` (Debian, Ubuntu, Fedora, RHEL, CentOS)

Sugere-se usar um subdiretório em vez da raiz do diretório HTML. Para criar um subdiretório e copiar os arquivos do frontend do Zabbix para dentro dele, execute os seguintes comandos, informando o diretório raiz junto ao nome do subdiretório que se pretende criar:

```
mkdir <htdocs>/zabbix
cd ui
cp -a . <htdocs>/zabbix
```

Para mais instruções quando usando idioma diferente do Inglês, consulte [Instalação de idiomas de frontend adicionais](#).

Instalação do frontend

Por favor, consulte a página [Instalação da interface web](#) para se informar sobre o assistente de instalação do Zabbix frontend.

3 Instalação do Java Gateway

A instalação do Java Gateway é exigida apenas se você pretende monitorar aplicações JMX. O Java Gateway é leve e não necessita de um banco de dados.

Para instalar a partir dos fontes, primeiro faça o **download** e então extraia o arquivo fonte.

Para compilar o Java Gateway, execute o script `./configure` com a opção `--enable-java`. É aconselhável que você especifique `--prefix` para direcionar a instalação para outro local diferente do padrão `/usr/local`, porque a instalação criará uma árvore de diretórios completa, e não apenas um simples executável.

```
$ ./configure --enable-java --prefix=$PREFIX
```

Para compilar e empacotar o Java Gateway em um arquivo JAR, execute `make`. Note que para este processo você necessitará dos executáveis `javac` e `jar` no seu path.

```
$ make
```

Agora você tem um arquivo `zabbix-java-gateway-$VERSION.jar` em `src/zabbix_java/bin`. Se você se sente confortável em executar o Java Gateway a partir de `src/zabbix_java` no diretório da distribuição, então você pode prosseguir às instruções de configuração e execução do **Java Gateway**. Caso contrário, certifique-se de possuir os privilégios necessários e execute `make install`.

```
$ make install
```

Prossiga até **Configuração inicial** para mais detalhes na configuração e execução do Java Gateway.

4 Instalando o serviço web Zabbix

A instalação do serviço web Zabbix só é necessária se você quiser usar **relatório agendado**.

Para instalar a partir de fontes, primeiro **download** e extraia o arquivo com as fontes .

Para compilar o serviço web Zabbix, execute o script `./configure` com opção `--enable-webservice` .

Note:

Uma versão **Go** configurada com ambiente 1.13+ é necessário para construir o serviço web Zabbix.

Execute `zabbix_web_service` na máquina onde o serviço web está instalado:

```
shell> zabbix_web_service
```

Prossiga para **configuração** para obter mais detalhes sobre como configurar a geração de relatórios agendados.

1 Criando Zabbix agent no Windows

Visão Geral

Esta seção demonstra como construir binários do agente Zabbix para Windows a partir de fontes com ou sem TLS..

Compilando OpenSSL

Os passos a seguir ajudarão a compilar OpenSSL a partir de fontes em MS Windows 10 (64-bit).

1. Para compilar OpenSSL em uma máquina com Windows, você precisará:
 1. C compilador (por exemplo, VS 2017 RC),
 2. NASM (<https://www.nasm.us/>),
 3. Perl (e.g. Strawberry Perl a partir de <http://strawberryperl.com/>),
 4. Perl module Text::Template (cpan Text::Template).
2. Obtenha as fontes OpenSSL a partir de <https://www.openssl.org/>. A versão OpenSSL 1.1.1 é usada.
3. Desempacote as fontes OpenSSL. Por exemplo, em `E:\openssl-1.1.1`.
4. Abra uma linha de comando Window. Por exemplo, o x64 Native Tools Command Prompt para VS 2017 RC.
5. Vá para o diretório de fontes do OpenSSL. Por exemplo, `E:\openssl-1.1.1`.
 1. Verifique que o NASM possa ser encontrado: `e:\openssl-1.1.1> nasm --version` NASM version 2.13.01 compiled on May 1 2017
6. Configure OpenSSL, por exemplo: `e:\openssl-1.1.1> perl E:\openssl-1.1.1\Configure VC-WIN64A no-shared no-capieng no-srp no-gost no-dgram no-dtls1-method no-dtls1_2-method --api=1.1.0 --prefix=C:\OpenSSL --openssldir=C:\OpenSSL-Win64-111-static`
 - Observe a opção 'não compartilhada': se 'não compartilhada' for usada, em seguida as bibliotecas estáticas OpenSSL `libcrypto.lib` e `libssl.lib` serão 'autossuficientes' e os resultados em binários Zabbix serão incluídos. Os próprios OpenSSL não necessitam de OpenSSL DLLs externos. Vantagem: Os binários Zabbix podem ser copiados para outras máquinas com Windows, sem bibliotecas OpenSSL. Desvantagem: quando uma nova versão de correção de bugs for lançada, o Zabbix agent necessita ser recompilado e reinstalado.
 - Se 'não compartilhado' não for usado, em seguida as bibliotecas estáticas `libcrypto.lib` e `libssl.lib` serão utilizadas em OpenSSL DLLs em tempo de execução. Vantagem: quando uma nova versão de correção de bugs for lançada, provavelmente você poderá atualizar apenas OpenSSL DLLs, sem recompilar o Zabbix agent. Desvantagem: copiar o Zabbix agent para outra máquina, requer também copiar OpenSSL DLLs.
7. Compile OpenSSL, execute os testes, instale: `e:\openssl-1.1.1> nmake` `e:\openssl-1.1.1> nmake test`
... All tests successful. Files=152, Tests=1152, 501 wallclock secs (0.67 usr + 0.61 sys = 1.28 CPU) Result: PASS `e:\openssl-1.1.1> nmake install_sw` `install_sw`
instalar apenas os componentes de software (por exemplo: bibliotecas, arquivos de cabeçalho, mas não a documentação). Caso você deseje instalar tudo, utilize o "nmake install".

Compilando PCRE

1. Baixe a biblioteca PCRE ou a PCRE2 (suportadas desde a versão Zabbix 6.0) (<https://pcre.org/>).
2. Extraia para diretório E:\pcre2-10.39.
3. Instale o CMake a partir de <https://cmake.org/download/>. Durante a instalação, selecione: e verifique que o cmake\bin esteja no seu atalho (versão testada 3.9.4).
4. Crie um diretório de compilação vazio, preferencialmente um subdiretório da origem dir. Por exemplo, E:\pcre2-10.39\build.
5. Abra um comando windows. Por exemplo, the x64 Native Tools Command Prompt para VS 2017 e naquele ambiente, execute cmake-gui. Não tente iniciar Cmake a partir do menu Widows Start, já que isso pode levar a erros.
6. Digite E:\pcre2-10.39 e E:\pcre2-10.39\build para a origem e construa diretórios, respectivamente.
7. Aperte o botão "Configurar".
8. Ao especificar o gerador para este projeto, selecione "NMake Makefiles".
9. Crie um novo diretório de instalação vazio. Por exemplo, E:\pcre2-10.39-install.
10. Em seguida, o GUI listará múltiplas opções de configuração. Certifique-se que as seguintes opções estejam selecionadas:
 - **PCRE_SUPPORT_UNICODE_PROPERTIES** ON
 - **PCRE_SUPPORT_UTF** ON
 - **CMAKE_INSTALL_PREFIX** E:\pcre2-10.39-install
11. Aperte em "Configurar" novamente. O botão adjacente "Generate" deverá estar ativo.
12. Aperte "Generate".
13. No evento em que ocorrer erros, é recomendável que você exclua o cache CMake antes de tentar repetir o processo de construção do CMake. No CMake GUI, o cache pode ser excluído selecionado o "File > Delete Cache".
14. O diretório de compilação agora deve conter um sistema de de compilação utilizável - Makefile.
15. Abra um comando Windows. Por exemplo, the x64 Native Tools Command Prompt para VS 2017 e navegue até o Makefile, mencionado acima.
16. Execute o comando NMake: E:\pcre2-10.39\build> nmake install

Compilando Zabbix

Os seguintes passos ajudarão você a compilar o Zabbix a partir das fontes no MS Windows 10 (64-bit). Ao compilar o Zabbix com/sem suporte TLS, a única diferença significativa é o passo 4.

1. Em uma máquina Linux, verifique a fonte a partir do git:\$ git clone https://git.zabbix.com/scm/zbx/zabbix.git
\$ cd zabbix \$./bootstrap.sh \$./configure --enable-agent --enable-ipv6 --prefix=`pwd`
\$ make dbschema \$ make dist
2. Copie e desempacote o arquivo. Por exemplo, zabbix-4.4.0.tar.gz, em uma máquina Windows.
3. Vamos assumir que as fontes estão em e:\zabbix-4.4.0. Abra um comando no windows. Por exemplo, e.g. the x64 Native Tools Command Prompt para VS 2017 RC. Vá para E:\zabbix-4.4.0\build\win32\project.
4. Compile zabbix_get, zabbix_sender and zabbix_agent.
 - sem TLS: E:\zabbix-4.4.0\build\win32\project> nmake /K PCREINCDIR=E:\pcre2-10.39-install\include PCRELIBDIR=E:\pcre2-10.39-install\lib
 - com TLS: E:\zabbix-4.4.0\build\win32\project> nmake /K -f Makefile_get TLS=openssl TLSINCDIR=C:\OpenSSL-Win64-111-static\include PCREINCDIR=E:\pcre2-10.39-install\include PCRELIBDIR=E:\pcre2-10.39-install\lib E:\zabbix-4.4.0\build\win32\project> nmake /K -f Makefile_sender TLS=openssl TLSINCDIR="C:\OpenSSL-Win64-111-static\include TLSLIBDIR="C:\OpenSSL-Win64-111-static\lib PCREINCDIR=E:\pcre2-10.39-install\include PCRELIBDIR=E:\pcre2-10.39-install\lib E:\zabbix-4.4.0\build\win32\project> nmake /K -f Makefile_agent TLS=openssl TLSINCDIR="C:\OpenSSL-Win64-111-static\include TLSLIBDIR="C:\OpenSSL-Win64-111-static\lib PCREINCDIR=E:\pcre2-10.39-install\include PCRELIBDIR=E:\pcre2-10.39-install\lib
5. Os binários novos estão localizados em e:\zabbix-4.4.0\bin\win64. Como o OpenSSL foi compilado com a opção 'não compartilhar', os binários do Zabbix contêm o OpenSSL internamente e podem ser copiados para outras máquina que não possuem o OpenSSL.

Compilando Zabbix com LibreSSL

O processo é similar a compilar com OpenSSL, mas agora você precisará fazer pequenas alterações nos arquivos localizados no diretório: build\win32\project:

- No Makefile_tls exclua /DHAVE_OPENSSL_WITH_PSK, por exemplo, encontre:

```
CFLAGS = $(CFLAGS) /DHAVE_OPENSSL /DHAVE_OPENSSL_WITH_PSK
```

e substitua por

```
CFLAGS = $(CFLAGS) /DHAVE_OPENSSL
```

- No Makefile_common.inc adicione /NODEFAULTLIB:LIBCMT por exemplo, encontre:

```
/MANIFESTUAC:"level='asInvoker' uiAccess='false'" /DYNAMICBASE:NO /PDB:${TARGETDIR}\$(TARGETNAME).pdb
```


e substitua por

```
/MANIFESTUAC:"level='asInvoker' uiAccess='false'" /DYNAMICBASE:NO /PDB:${TARGETDIR}\$(TARGETNAME).pdb /NOD
```

Construindo Agente Zabbix 2 no Windows

Visão geral

Esta seção demonstra como construir o Zabbix agent 2 (Windows) a partir de fontes.

Instalando o compilador MinGW

1. Download MinGW-w64 with SJLJ (set jump/long jump) Manipulação de exceção e tópicos do Windows (for example x86_64-8.1.0-release-win32-sjlj-rt_v6-rev0.7z)
2. Extract and move to c:\mingw
3. Setup environmental variable

```
@echo off
set PATH=%PATH%;c:\mingw\bin
cmd
```

Ao compilar, use o prompt do Windows em vez do terminal MSYS fornecido por MinGW

Compilando bibliotecas de desenvolvimento PCRE

As instruções a seguir irão compilar e instalar o PCRE de 64 bits bibliotecas em c:\dev\pcre e bibliotecas de 32 bits em c:\dev\pcre32:

1. Baixe a biblioteca PCRE versão 8.XX em pcre.org (<http://ftp.pcre.org/pub/pcre/>) e extrair
2. Abra cmd e navegue até as fontes extraídas

Build 64bit PCRE

1. Delete old configuration/cache if exists:

```
del CMakeCache.txt
rmdir /q /s CMakeFiles
```

2. Run cmake (CMake can be installed from <https://cmake.org/download/>):

```
cmake -G "MinGW Makefiles" -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-O2 -g" -DCMAKE_CXX_FLAGS="-O2 -g" -DCM
```

3. Next, run:

```
mingw32-make clean
mingw32-make install
```

Build 32bit PCRE

1. Run:

```
mingw32-make clean
```

2. Delete CMakeCache.txt:

```
del CMakeCache.txt
rmdir /q /s CMakeFiles
```

3. Run cmake:

```
cmake -G "MinGW Makefiles" -DCMAKE_C_COMPILER=gcc -DCMAKE_C_FLAGS="-m32 -O2 -g" -DCMAKE_CXX_FLAGS="-m32 -C
```

4. Next, run:

```
mingw32-make install
```

Instalando bibliotecas de desenvolvimento OpenSSL

1. Download versões 32 e 64 bit através de <https://curl.se/windows/>
2. Extraia os arquivos em c:\dev\openssl32 e c:\dev\openssl directories accordingly.
3. Depois disso, remova *.dll.a (bibliotecas de wrapper de chamadas dll) já que o MinGW os prioriza antes das bibliotecas estáticas.

Compilando agente Zabbix 2

32 bit

Abra o ambiente MinGW (prompt de comando do Windows) e navegue até build/mingw diretório de fonte Zabbix .

Execute:

```
mingw32-make clean
mingw32-make ARCH=x86 PCRE=c:\dev\pcre32 OPENSsl=c:\dev\openssl32
```

64 bit

Abra o ambiente MinGW (prompt de comando do Windows) e navegue até build/mingw diretório de fonte Zabbix .

Execute:

```
mingw32-make clean
mingw32-make PCRE=c:\dev\pcre OPENSsl=c:\dev\openssl
```

Note:

Ambas as versões de 32 e 64 bits podem ser criadas em uma plataforma 64-bit, mas apenas a versão 32-bit pode criar em uma plataforma 32-bit. Ao trabalhar na plataforma de 32 bits, siga as mesmas etapas para 64 bits para a plataforma versão 64 bits.

Criando o agente Zabbix no macOS

Visão geral

Esta seção demonstra como construir os binários do agente Zabbix para macOS a partir de fontes com ou sem TLS.

Pré-requisitos

Você precisará de ferramentas de desenvolvedor de linha de comando (Xcode não é necessário), Automake, pkg-config e PCRE (v8.x) ou PCRE2 (v10.x). Se você deseja criar os binários do agente com TLS, precisará também do OpenSSL ou GnuTLS.

Para instalar o Automake e o pkg-config, você precisará de um gerenciador de pacotes Homebrew de <https://brew.sh/>. Para instalar, abra o terminal e execute o seguinte comando:

```
$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

Em seguida, instale o Automake e o pkg-config:

```
$ brew install automake
$ brew install pkg-config
```

A preparação das bibliotecas do PCRE, OpenSSL e GnuTLS depende da maneira como eles serão vinculados ao agente.

Se você pretende executar os binários do agente em uma máquina macOS que possui estas bibliotecas, você pode utilizar bibliotecas pré-compiladas fornecidas pelo Homebrew. Normalmente, são máquinas macOS que usam o Homebrew para criar binários do agente Zabbix ou para outros fins.

Se os binários do agente forem utilizados em máquinas macOS que não possuem a versão compartilhada das bibliotecas, você deve compilar bibliotecas estáticas a partir de fontes e vincular o agente Zabbix com eles.

Construindo binários do agente com bibliotecas compartilhadas

Instalar o PCRE2 (substitua pcre2 por pcre nos comandos abaixo, caso necessário):

```
$ brew install pcre2
```

Ao construir com TLS, instale OpenSSL e/ou GnuTLS::

```
$ brew install openssl
$ brew install gnutls
```

Baixar o fonte do Zabbix:

```
$ git clone https://git.zabbix.com/scm/zbx/zabbix.git
```

Construir o agente sem TLS:

```
$ cd zabbix
$ ./bootstrap.sh
$ ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6
$ make
$ make install
```

Construir o agente com OpenSSL:

```
$ cd zabbix
$ ./bootstrap.sh
$ ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-openssl=/usr/local/opt
$ make
$ make install
```

Construir o agente com GnuTLS:

```
$ cd zabbix-source/
$ ./bootstrap.sh
$ ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-gnutls=/usr/local/opt
$ make
$ make install
```

Construindo binários de agente com bibliotecas estáticas sem TLS

Vamos assumir que as bibliotecas estáticas do PCRE serão instaladas em `$HOME/static-libs`. Utilizaremos o PCRE2 10.39.

```
$ PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
```

Faça o download e construa o PCRE com suporte a propriedades Unicode:

```
$ mkdir static-libs-source
$ cd static-libs-source
$ curl --remote-name https://github.com/PhilipHazel/pcre2/releases/download/pcre2-10.39/pcre2-10.39.tar.gz
$ tar xf pcre2-10.39.tar.gz
$ cd pcre2-10.39
$ ./configure --prefix="$PCRE_PREFIX" --disable-shared --enable-static --enable-unicode-properties
$ make
$ make check
$ make install
```

Faça o download do fonte do Zabbix e construa o agente:

```
$ git clone https://git.zabbix.com/scm/zbx/zabbix.git
$ cd zabbix
$ ./bootstrap.sh
$ ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFIX"
$ make
$ make install
```

Construindo binários do agente com bibliotecas estáticas com o OpenSSL

Ao construir o OpenSSL, é recomendado executar `make test` após construção com sucesso. Mesmo que a construção tenha sido bem-sucedida, os testes às vezes falham. Se este for o caso, os problemas devem ser pesquisados e resolvidos antes de continuar.

Vamos assumir que as bibliotecas estáticas PCRE e OpenSSL serão instaladas em `$HOME/static-libs`. Utilizaremos o PCRE2 10.39 e OpenSSL 1.1.1a.

```
$ PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
$ OPENSSL_PREFIX="$HOME/static-libs/openssl-1.1.1a"
```

Vamos construir as bibliotecas estáticas em `static-libs-source`:

```
$ mkdir static-libs-source
$ cd static-libs-source
```

Faça o download e construa o PCRE com suporte a propriedades Unicode:

```
$ curl --remote-name https://github.com/PhilipHazel/pcre2/releases/download/pcre2-10.39/pcre2-10.39.tar.gz
$ tar xf pcre2-10.39.tar.gz
$ cd pcre2-10.39
$ ./configure --prefix="$PCRE_PREFIX" --disable-shared --enable-static --enable-unicode-properties
$ make
$ make check
$ make install
$ cd ..
```

Faça o download e construa o OpenSSL:

```
$ curl --remote-name https://www.openssl.org/source/openssl-1.1.1a.tar.gz
$ tar xf openssl-1.1.1a.tar.gz
```

```
$ cd openssl-1.1.1a
$ ./Configure --prefix="$OPENSSL_PREFIX" --openssldir="$OPENSSL_PREFIX" --api=1.1.0 no-shared no-capieng n
$ make
$ make test
$ make install_sw
$ cd ..
```

Faça o download do fonte do Zabbix e construa o agente:

```
$ git clone https://git.zabbix.com/scm/zbx/zabbix.git
$ cd zabbix
$ ./bootstrap.sh
$ ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFI
$ make
$ make install
```

Construindo binários de agentes com bibliotecas estáticas com GnuTLS

O GnuTLS depende do back-end de criptografia Nettle e da biblioteca aritmética GMP. Em vez de usar a biblioteca GMP completa, este guia usará o mini-gmp, que é incluído na Nettle.

Ao compilar GnuTLS e Nettle, é recomendado executar `make check` após a compilação bem-sucedida. Mesmo que a construção tenha sido bem-sucedida, os testes às vezes falha. Se este for o caso, os problemas devem ser pesquisados e resolvido antes de continuar.

Vamos assumir que as bibliotecas estáticas do PCRE, Nettle e GnuTLS serão instaladas em `$HOME/static-libs`. Utilizaremos o PCRE2 10.39, Nettle 3.4.1 e GnuTLS 3.6.5.

```
$ PCRE_PREFIX="$HOME/static-libs/pcre2-10.39"
$ NETTLE_PREFIX="$HOME/static-libs/nettle-3.4.1"
$ GNUTLS_PREFIX="$HOME/static-libs/gnutls-3.6.5"
```

Vamos construir bibliotecas estáticas em `static-libs-source`:

```
$ mkdir static-libs-source
$ cd static-libs-source
```

Faça o download e construa o Nettle:

```
$ curl --remote-name https://ftp.gnu.org/gnu/nettle/nettle-3.4.1.tar.gz
$ tar xf nettle-3.4.1.tar.gz
$ cd nettle-3.4.1
$ ./configure --prefix="$NETTLE_PREFIX" --enable-static --disable-shared --disable-documentation --disabl
$ make
$ make check
$ make install
$ cd ..
```

Faça o download e construa o GnuTLS:

```
$ curl --remote-name https://www.gnupg.org/ftp/gcrypt/gnutls/v3.6/gnutls-3.6.5.tar.xz
$ tar xf gnutls-3.6.5.tar.xz
$ cd gnutls-3.6.5
$ PKG_CONFIG_PATH="$NETTLE_PREFIX/lib/pkgconfig" ./configure --prefix="$GNUTLS_PREFIX" --enable-static --d
$ make
$ make check
$ make install
$ cd ..
```

Faça o download do fonte do Zabbix e construa o agente:

```
$ git clone https://git.zabbix.com/scm/zbx/zabbix.git
$ cd zabbix
$ ./bootstrap.sh
$ CFLAGS="-Wno-unused-command-line-argument -framework Foundation -framework Security" \
> LIBS="-lgnutls -lhogweed -lnettle" \
> LDFLAGS="-L$GNUTLS_PREFIX/lib -L$NETTLE_PREFIX/lib" \
> ./configure --sysconfdir=/usr/local/etc/zabbix --enable-agent --enable-ipv6 --with-libpcre2="$PCRE_PREFI
$ make
$ make install
```

4 Instalação a partir de pacotes

A partir do repositório oficial do Zabbix

A Zabbix SIA fornece pacotes oficiais RPM e DEB para::

- [Red Hat Enterprise Linux](#)
- [Debian/Ubuntu/Raspbian](#)
- [SUSE Linux Enterprise Server](#)

Os arquivos de pacote para repositórios yum/dnf, apt e zypper para várias distribuições de sistema operacional estão disponíveis em repo.zabbix.com.

Observe que, embora algumas distribuições de sistema operacional (especialmente, distribuições baseadas em Debian) forneçam seus próprios pacotes Zabbix, esses pacotes não são suportados pelo Zabbix. Os pacotes Zabbix fornecidos por terceiros podem estar desatualizados e podem não conter as últimas funcionalidades e correções de bugs. É recomendável usar apenas os pacotes oficiais de repo.zabbix.com. Se você já usou pacotes Zabbix não oficiais, consulte as notas sobre atualização dos pacotes Zabbix [upgrading Zabbix packages from OS repositories](#).

1 Sistema operacional Red Hat Enterprise Linux

Visão geral

Os pacotes oficiais Zabbix 6.0 LTS para Red Hat Enterprise Linux e Oracle Linux estão disponíveis em [Zabbix website](#).

Pacotes estão disponíveis com suporte a banco de dados MySQL/PostgreSQL e servidor web Apache/Nginx.

Pacotes Zabbix agent e utilitários Zabbix get e Zabbix sender estão disponíveis no Repositório Oficial Zabbix para [RHEL 9](#), [RHEL 8](#), [RHEL 7](#), [RHEL 6](#), e [RHEL 5](#).

O Repositório Oficial Zabbix também fornece pacotes fping, iksemel e libssh2. Esses pacotes estão localizados no diretório [non-supported](#).

Attention:

O repositório EPEL para EL9 também fornece os pacotes Zabbix. Se ambos estiverem instalados (repositório oficial Zabbix e EPEL repositórios), os pacotes do Zabbix no EPEL **must be** (devem ser) excluídos, adicionando a seguinte cláusula ao arquivo de configuração do repositório EPEL `/etc/yum.repos.d/`:

```
[epel]
...
excludepkgs=zabbix*
```

Notas de instalação

Consulte as instruções de instalação [installation instructions](#) por plataforma na página de download para:

- instalar repositório
- instalar servidor/agente/frontend
- criar banco de dados inicial, importar dados inicial
- configurar bancos dados para servidor Zabbix
- configurar PHP para Zabbix frontend
- iniciar processos servidor/agente
- configurar Zabbix frontend

Se você deseja executar o agente Zabbix como raiz, consulte em [Running agent as root](#).

O processo de serviço web do Zabbix, que é usado para [scheduled reportgeneration](#), requer o navegador Google Chrome. O navegador não está incluso nos pacotes e precisa ser instalado manualmente.

Importar dados com Timescale DB

Com o TimescaleDB, além do comando de importação para o PostgreSQL, também execute:

```
cat /usr/share/zabbix-sql-scripts/postgresql/timescaledb.sql | sudo -u zabbix psql zabbix
```

Warning:

O TimescaleDB somente é suportado com servidor Zabbix.

PHP 7.2

O Zabbix frontend requer a versão PHP **7.2 or newer**.

Configuração SELinux

Zabbix utiliza a comunicação socket-based inter-process. Nos sistemas em que o SELinux está habilitado, pode ser necessário adicionar regras de SELinux para permitir que o Zabbix crie e use soquetes de domínio UNIX no diretório SocketDir. Atualmente, os arquivos de soquetes são usados pelo servidor (alerta, pré-processamento, IPMI) e proxy (IPMI). Os arquivos de soquete são persistentes, isso significa que eles estão presentes enquanto o processo está sendo executado.

Tendo o status SELinux habilitado no modo de aplicação, você precisa executar os seguintes comandos para habilitar a comunicação entre o Zabbix frontend e o servidor:

RHEL 7 and later:

```
setsebool -P httpd_can_connect_zabbix on
```

Se o banco de dados for acessível pela rede (incluir 'localhost' em caso de PostgreSQL), você precisa autorizar a conexão do Zabbix frontend com a bases de dados também:

```
setsebool -P httpd_can_network_connect_db on
```

RHEL prior to 7:

```
setsebool -P httpd_can_network_connect on
setsebool -P zabbix_can_network on
```

Uma vez que a configuração de frontend e SELinux estiver feita, reinicie o servidor Apache:

```
service httpd restart
```

Além disso, o Zabbix disponibiliza o pacote zabbix-selinux-policy como parte dos pacotes RPM de origem para [RHEL 8](#) e [RHEL 7](#). Este pacote fornece uma política padrão básica para o SELinux e permite que os componentes do Zabbix funcionem out-of-the-box (fora da caixa), permitindo que o Zabbix crie e use soquetes e habilite a conexão do https com o PostgreSQL (usado pelo frontend).

O arquivo de origem zabbix_policy.te contém as seguintes regras:

```
module zabbix_policy 1.2;

require {
    type zabbix_t;
    type zabbix_port_t;
    type zabbix_var_run_t;
    type postgresql_port_t;
    type httpd_t;
    class tcp_socket name_connect;
    class sock_file { create unlink };
    class unix_stream_socket connectto;
}

#===== zabbix_t =====
allow zabbix_t self:unix_stream_socket connectto;
allow zabbix_t zabbix_port_t:tcp_socket name_connect;
allow zabbix_t zabbix_var_run_t:sock_file create;
allow zabbix_t zabbix_var_run_t:sock_file unlink;
allow httpd_t zabbix_port_t:tcp_socket name_connect;

#===== httpd_t =====
allow httpd_t postgresql_port_t:tcp_socket name_connect;
```

Este pacote foi criado para evitar que os usuários desativem o SELinux por devido à complexidade da configuração. O pacote contém a política padrão que é o suficiente para acelerar a implantação e configuração do Zabbix. Para o nível máximo de segurança, é recomendável configurar as definições personalizadas do SELinux.

Instalação do proxy

Uma vez que o repositório requisitado é adicionado, você pode instalar o proxy do Zabbix executando:

```
dnf install zabbix-proxy-mysql zabbix-sql-scripts
```

Substitua 'mysql' nos comandos com 'pgsql' para usar PostgreSQL, ou com 'sqlite3' para utilizar SQLite3 (somente proxy).

O pacote 'zabbix-sql-scripts' contém esquemas de banco de dados para todos os sistemas de gerenciamento de bancos de dados suportados, tanto para o servidor Zabbix quanto para o proxy Zabbix, e será utilizado para importação de dados.

Criando base de dados

[Criar] (/manual/appendix/install/db_scripts) uma base de dados separada para o Zabbix proxy. O servidor Zabbix e o proxy Zabbix não podem usar a mesma base de dados. Se elas são instaladas no mesmo host, a base de dados do proxy deve ter um nome diferente.

Importando dados

Importar esquema inicial:

```
cat /usr/share/zabbix-sql-scripts/mysql/proxy.sql | mysql -uzabbix -p zabbix
```

Para proxy com PostgreSQL (ou SQLite):

```
cat /usr/share/zabbix-sql-scripts/postgresql/proxy.sql | sudo -u zabbix psql zabbix
cat /usr/share/zabbix-sql-scripts/sqlite3/proxy.sql | sqlite3 zabbix.db
```

Configurar banco de dados para o proxy Zabbix

Editar zabbix_proxy.conf:

```
# vi /etc/zabbix/zabbix_proxy.conf
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=<password>
```

No DBName para o proxy Zabbix, utilize um banco de dados separado do servidor Zabbix.

No DBPassword, utilize a senha do banco de dados Zabbix para o MySQL; a senha do usuário do PostgreSQL para PostgreSQL.

Utilize DBHost= com PostgreSQL. Você pode optar por manter a configuração padrão DBHost=localhost (ou um endereço de IP), mas isso fará com que o PostgreSQL utilize um soquete de rede para se conectar com o Zabbix. Consulte [SELinux configuration](#) para instruções.

Iniciando o processo do proxy Zabbix

Para iniciar um processo do proxy Zabbix e fazê-lo iniciar automaticamente durante a inicialização do sistema:

```
service zabbix-proxy start
systemctl enable zabbix-proxy
```

Configuração Frontend

O proxy Zabbix não possui um frontend; ele se comunica somente com o servidor Zabbix.

Instalação do Java gateway

É necessário instalar o [Java gateway](#) somente se você seja monitorar as aplicações JMX. O gateway Java é leve e não requer um banco de dados.

Uma vez que o repositório necessário é adicionado, você pode instalar o Zabbix Java gateway executando:

```
dnf install zabbix-java-gateway
```

Prossiga para [setup](#) para mais detalhes da configuração e execução do Java gateway.

Instalando pacotes debuginfo

Os pacotes Debuginfo atualmente estão disponíveis para as versões 7, 6 e 5 do RHE.

Para habilitar o repositório debuginfo, editar o arquivo /etc/yum.repos.d/zabbix.repo. Altere enabled=0 para enabled=1 para o repositório zabbix-debuginfo.

```
[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo - $basearch
baseurl=http://repo.zabbix.com/zabbix/5.5/rhel/7/$basearch/debuginfo/
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
gpgcheck=1
```

Isso permitirá que você instale o pacote zabbix-debuginfo.

```
dnf install zabbix-debuginfo
```

Esse pacote único contém informações debug para todos os componentes binários do Zabbix.

2 Debian/Ubuntu/Raspbian

Visão geral

Os pacotes oficiais do Zabbix 6.0 LTS para Debian, Ubuntu, e Raspberry Pi OS (Raspbian) estão disponíveis no [Zabbix website](#).

Os pacotes estão disponíveis para banco de dados MySQL/PostgreSQL e para servidor web Apache/Nginx web.

Notas sobre a instalação

Ver as instruções por plataforma [installation](#) na página de download para:

- instalar repositório
- instalar servidor/agente/frontend
- criar banco de dados inicial, importar dado inicial
- configurar banco de dados para o servidor Zabbix
- configurar PHP para frontend no Zabbix
- iniciar processos do servidor/agente
- configurar frontend no Zabbix

Se você quiser executar o Zabbix agent como root, consulte [running agent as root](#).

O processo de serviço de web do Zabbix utilizado [scheduled report generation](#), requer o navegador Google Chrome. O navegador não está incluído nos pacotes e deve ser instalado manualmente.

Importar dados com Timescale DB

Com o TimescaleDB, além do comando de importação para o PostgreSQL, também execute:

```
cat /usr/share/zabbix-sql-scripts/postgresql/timescaledb.sql | sudo -u zabbix psql zabbix
```

Warning:

O TimescaleDB somente é suportado com servidor Zabbix.

PHP 7.2

O frontend Zabbix requer PHP na versão **7.2 or newer** a partir do Zabbix 5.0.

Consulte as instruções [instructions](#) para instalar frontend Zabbix em distribuições com versões do PHP inferiores a 7.2.

Configuração SELinux

Veja [SELinux configuration](#) para RHEL/CentOS.

Após a configuração do frontend e do SELinux, reinicie o Apache servidor web:

```
# service apache2 restart
```

Instalação Proxy

Assim que o repositório necessário for adicionado, você pode instalar o proxy Zabbix por Executando:

```
# apt install zabbix-proxy-mysql
```

Substitua 'mysql' no comando por 'pgsql' para usar PostgreSQL, ou com 'sqlite3' para usar SQLite3.

Criando banco de dados

[Create](#) um banco de dados separado para Proxy Zabbix.

O servidor Zabbix e o proxy Zabbix não podem usar o mesmo banco de dados. Se eles são instalado no mesmo host, o banco de dados proxy deve ter um diferente nome.

Importando dados

Importar esquema inicial:

```
cat /usr/share/zabbix-sql-scripts/mysql/proxy.sql | mysql -uzabbix -p zabbix
```

Para proxy com PostgreSQL (ou SQLite):

```
cat /usr/share/zabbix-sql-scripts/postgresql/proxy.sql | sudo -u zabbix psql zabbix
cat /usr/share/zabbix-sql-scripts/sqlite3/proxy.sql | sqlite3 zabbix.db
```


Configurar banco de dados para Zabbix proxy

Edit zabbix_proxy.conf:

```
# vi /etc/zabbix/zabbix_proxy.conf
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=<password>
```

Em DBName para Zabbix proxy, use um banco de dados separado do servidor Zabbix.

Em DBPassword, use a senha do banco de dados Zabbix para MySQL; Usuário PostgreSQL senha para PostgreSQL.

Use DBHost = com PostgreSQL. Você pode querer manter o padrão definir DBHost = localhost (ou um endereço IP), mas isso faria O PostgreSQL usa um soquete de rede para se conectar ao Zabbix. Consulte o [respective section](#) para RHEL/CentOS para obter instruções.

Iniciando o processo de Zabbix proxy

Para iniciar um processo de Zabbix proxy e fazê-lo iniciar na inicialização do sistema:

```
# systemctl restart zabbix-proxy
# systemctl enable zabbix-proxy
```

Configuração de front-end

Um Zabbix proxy não tem front-end; ele se comunica com o Zabbix servidor apenas.

Instalação do Java gateway

É necessário instalar [Java gateway](#) somente se você deseja monitorar aplicativos JMX. O gateway Java é leve e não requer um banco de dados.

Assim que o repositório necessário for adicionado, você pode instalar o Zabbix Java gateway executando:

```
# apt install zabbix-java-gateway
```

Prossiga para [setup](#) para mais detalhes sobre como configurar e executar o Java gateway.

3 SUSE Linux Enterprise Server

Visão geral

Os pacotes oficiais LTS Zabbix 6.0 para SUSE Linux Enterprise Server estão disponíveis em [Zabbix website](#).

Os pacotes Zabbix agent e utilidades Zabbix get e Zabbix sender estão disponíveis no repositório do Zabbix [SLES 15](#) e [SLES 12](#).

Note:

O Verify CA [encryption mode](#) não funciona em SLES 12 (todas as versões menores de OS) com MySQL devido a bibliotecas antigas do MySQL.

Adicionando repositório Zabbix

Instale o pacote de configuração do repositório. Este pacote contém arquivos de configuração do YUM (gerenciador de pacotes).

SLES 15:

```
# rpm -Uvh --nosignature https://repo.zabbix.com/zabbix/6.0/sles/15/x86_64/zabbix-release-6.0-1.sles15.noarch.rpm
# zypper --gpg-auto-import-keys refresh 'Zabbix Official Repository'
```

SLES 12:

```
# rpm -Uvh --nosignature https://repo.zabbix.com/zabbix/6.0/sles/12/x86_64/zabbix-release-6.0-1.sles12.noarch.rpm
# zypper --gpg-auto-import-keys refresh 'Zabbix Official Repository'
```

Por favor, note que o processo de web service do Zabbix usado para a [geração de relatório agendado](#) requer o navegador Google Chrome. O navegador não está incluso nos pacotes e deve ser instalado manualmente.

Instalação Server/Frontend/Agent

Para instalar o Zabbix Server/Frontend/Agent com suporte ao MySQL:

```
# zypper install zabbix-server-mysql zabbix-web-mysql zabbix-apache-conf zabbix-agent
```

Substitua 'apache' no comando por 'nginx' se estiver usando pacote para Nginx Web Server. Veja também: [Configuração do Nginx para Zabbix no SLES 12/15](#).

Substitua 'zabbix-agent' por 'zabbix-agent2' nestes comandos se estiver usando Zabbix Agent 2 (apenas SLES 15 SP1+).

Para instalar o Zabbix Proxy com suporte ao MySQL:

```
# zypper install zabbix-proxy-mysql
```

Substitua 'mysql' nos comandos por 'pgsql' para usar PostgreSQL.

Criando Banco de Dados

Para os processos do Zabbix **Server** e **Proxy** um banco de dados é exigido. Ele não é necessário para executar o Zabbix **Agent**.

Warning:

Bancos de dados isolados são necessários para o Zabbix Server e Zabbix Proxy; eles não podem utilizar o mesmo banco de dados. Portanto, se eles estiverem instalados na mesma máquina, seus bancos de dados devem ser criados com nomes diferentes!

Crie os bancos de dados usando as instruções disponíveis para **MySQL** ou **PostgreSQL**.

Importando dados

Agora importe o schema inicial e os dados para o **server** com MySQL:

```
# zcat /usr/share/doc/packages/zabbix-sql-scripts/mysql/create.sql.gz | mysql -uzabbix -p zabbix
```

Será solicitado que você informe a senha para o banco de dados recém-criado.

Com PostgreSQL:

```
# zcat /usr/share/doc/packages/zabbix-sql-scripts/postgresql/create.sql.gz | sudo -u zabbix psql zabbix
```

Com TimescaleDB, em adição ao comando anterior, também execute:

```
# zcat /usr/share/doc/packages/zabbix-sql-scripts/postgresql/timescaledb.sql.gz | sudo -u <username> psql
```

Warning:

TimescaleDB é suportado apenas com Zabbix server.

Para o proxy, importe o schema inicial:

```
# zcat /usr/share/doc/packages/zabbix-sql-scripts/mysql/schema.sql.gz | mysql -uzabbix -p zabbix
```

Para proxy com PostgreSQL:

```
# zcat /usr/share/doc/packages/zabbix-sql-scripts/postgresql/schema.sql.gz | sudo -u zabbix psql zabbix
```

Configure o banco de dados para Zabbix Server/Proxy

Edite /etc/zabbix/zabbix_server.conf (e zabbix_proxy.conf) para usar seu respectivo banco de dados. Por exemplo:

```
# vi /etc/zabbix/zabbix_server.conf
DBHost=localhost
DBName=zabbix
DBUser=zabbix
DBPassword=<password>
```

Em DBPassword use a senha do banco de dados do Zabbix no MySQL; senha do usuário PostgreSQL se estiver utilizando banco de dados PostgreSQL.

Use DBHost= com PostgreSQL. Você poderia querer manter a configuração padrão DBHost=localhost (ou um endereço IP), mas isto faria o PostgreSQL usar um socket de rede para se conectar ao Zabbix.

Configuração do Zabbix Frontend

Dependendo do Web Server utilizado (Apache/Nginx) edite o arquivo de configuração correspondente para o Zabbix Frontend:

- Para o Apache encontre o arquivo de configuração em /etc/apache2/conf.d/zabbix.conf. Algumas definições de PHP já estão configuradas. Mas é necessário descomentar a definição "date.timezone" e [informar o timezone adequado](#) para você.

```

php_value max_execution_time 300
php_value memory_limit 128M
php_value post_max_size 16M
php_value upload_max_filesize 2M
php_value max_input_time 300
php_value max_input_vars 10000
php_value always_populate_raw_post_data -1
# php_value date.timezone Europe/Riga

```

- O pacote zabbix-nginx-conf instala um servidor Nginx separado para o Zabbix Frontend. Seu arquivo de configuração está localizado em `/etc/nginx/conf.d/zabbix.conf`. Para o Zabbix Frontend funcionar, é necessário descomentar e configurar as diretivas `listen` e/ou `server_name`.

```

# listen 80;
# server_name example.com;

```

- O Zabbix usa seu próprio pool de conexão php-fpm dedicado com Nginx:

Ser arquivo de configuração está localizado em `/etc/php7/fpm/php-fpm.d/zabbix.conf`. Algumas definições de PHP já estão configuradas. Mas é necessário descomentar a definição `"date.timezone"` e [informar o timezone adequado](#) para você.

```

php_value[max_execution_time] = 300
php_value[memory_limit] = 128M
php_value[post_max_size] = 16M
php_value[upload_max_filesize] = 2M
php_value[max_input_time] = 300
php_value[max_input_vars] = 10000
; php_value[date.timezone] = Europe/Riga

```

Agora você está pronto para prosseguir com os [passos de instalação do Frontend](#) que lhe permitirão o acesso ao seu Zabbix recém-instalado.

Note que um Zabbix Proxy não tem um frontend; ele se comunica apenas com o Zabbix Server.

Iniciando o processo Zabbix Server/Agent

Inicie os processos do Zabbix Server e Agent e certifique-se de que iniciem com o boot do sistema.

Com Apache Web Server:

```

# systemctl restart zabbix-server zabbix-agent apache2 php-fpm
# systemctl enable zabbix-server zabbix-agent apache2 php-fpm

```

Substitua 'apache2' por 'nginx' para Nginx Web Server.

Instalando pacotes de debuginfo

Para habilitar o repositório de debuginfo edite o arquivo `/etc/zypp/repos.d/zabbix.repo`. Altere `enabled=0` para `enabled=1` no repositório `zabbix-debuginfo`.

```

[zabbix-debuginfo]
name=Zabbix Official Repository debuginfo
type=rpm-md
baseurl=http://repo.zabbix.com/zabbix/4.5/sles/15/x86_64/debuginfo/
gpgcheck=1
gpgkey=http://repo.zabbix.com/zabbix/4.5/sles/15/x86_64/debuginfo/repodata/repomd.xml.key
enabled=0
update=1

```

Isto permitirá que você instale pacotes `zabbix-<component>-debuginfo`.

4 Instalação agente Windows a partir do MSI

Visão Geral

O agente do Zabbix para Windows pode ser instalado a partir dos pacotes de instalador MSI do Windows (32 bits ou 64 bits) disponíveis para [download](#).

O requisito mínimo para a instalação do MSI do agente Zabbix 2 é o Windows 10 32 bits/Servidor 2016.

As utilidades Zabbix get e sender também podem ser instaladas, seja junto com o Zabbix agent/agent 2 ou separadamente.

Um pacote de 32 bits não pode ser instalado em um Windows de 64 bits.

Todos os pacotes vêm com suporte TLS, no entanto, configurar o TLS é opcional.

Tanto a instalação baseada em interface de usuário quanto a baseada em linha de comando são suportadas.

Note:

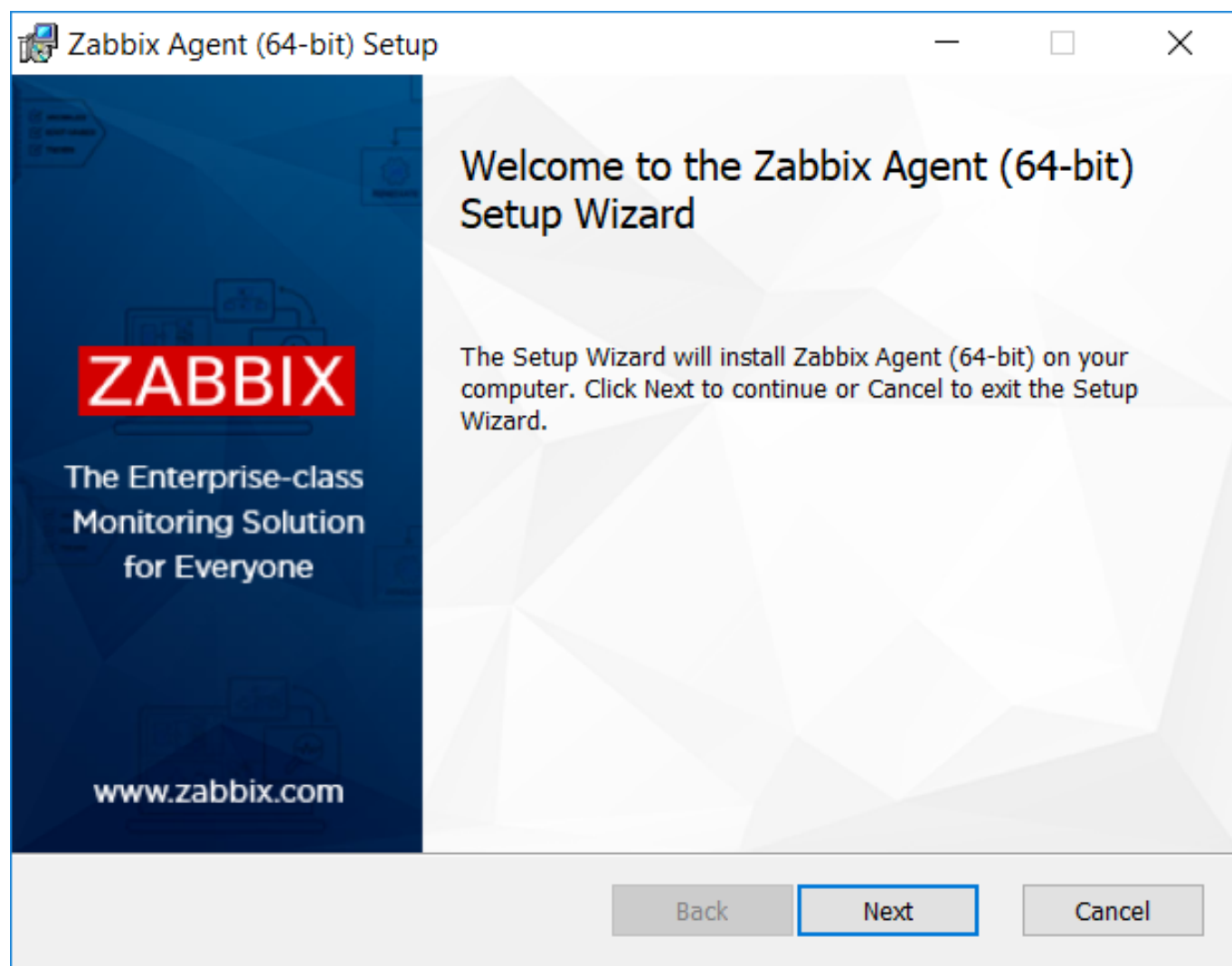
Embora a instalação do Zabbix a partir dos pacotes de instalador MSI seja totalmente suportada, é recomendável instalar pelo menos o Microsoft .NET Framework 2 para um tratamento adequado de erros. Consulte [Microsoft Download .NET Framework](#).

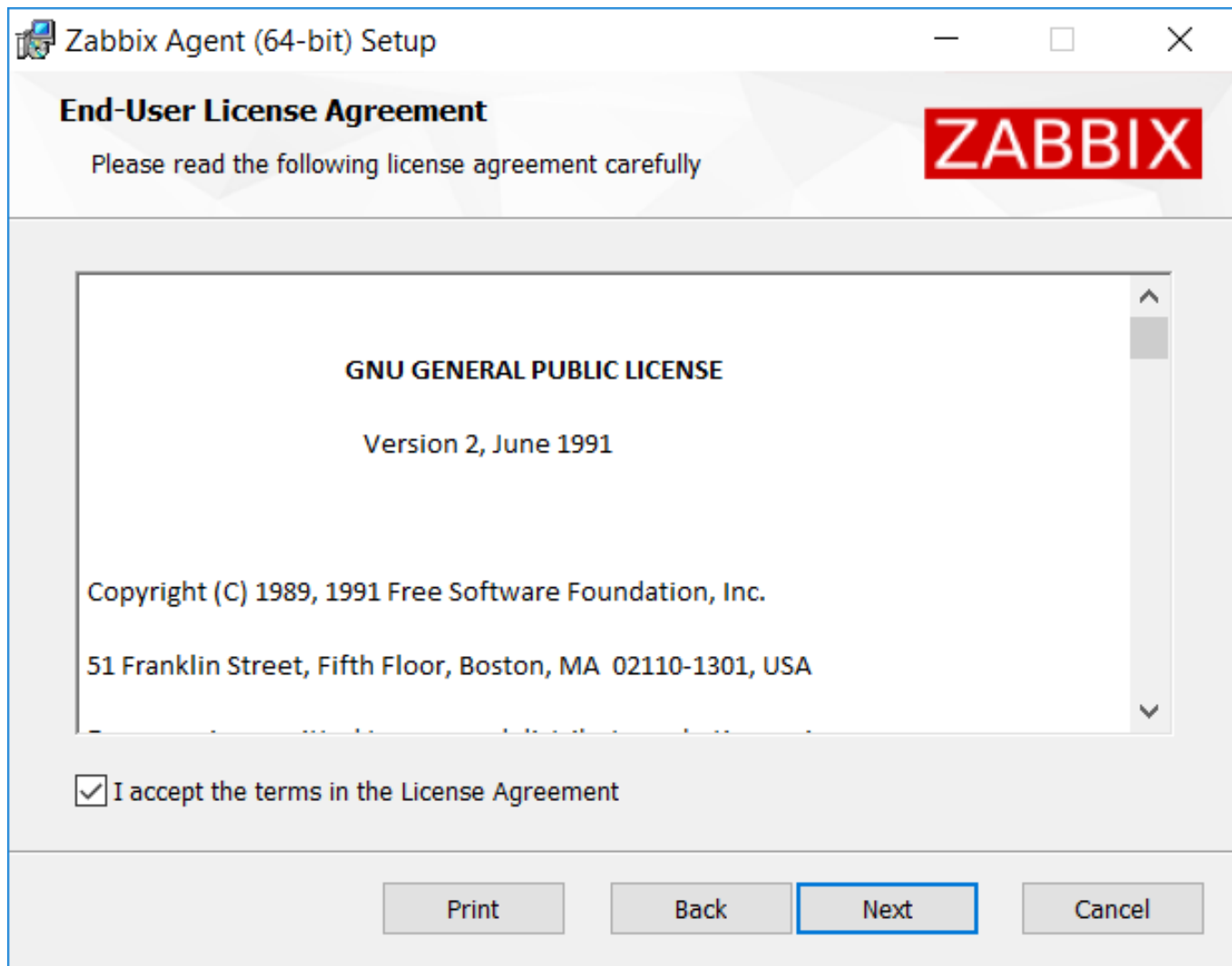
Attention:

É recomendável usar os caminhos padrão fornecidos pelo instalador, pois o uso de caminhos personalizados sem as permissões adequadas pode comprometer a segurança da instalação.

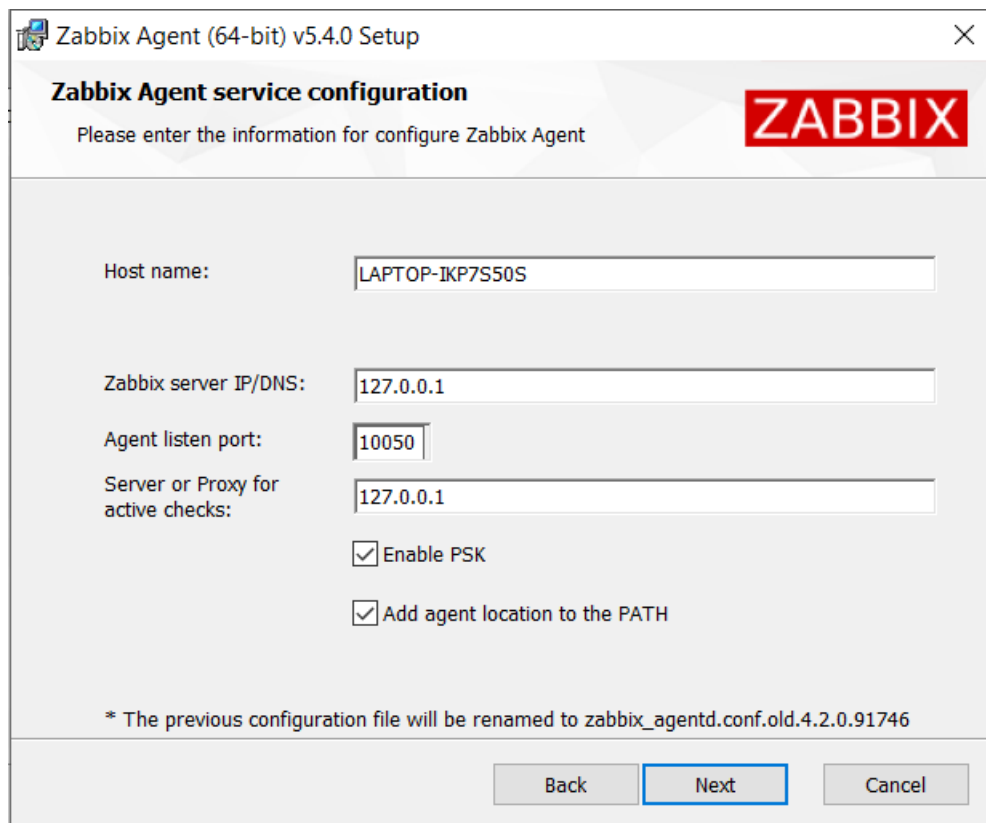
Etapas de instalação

Para instalar, clique duas vezes no arquivo MSI baixado.



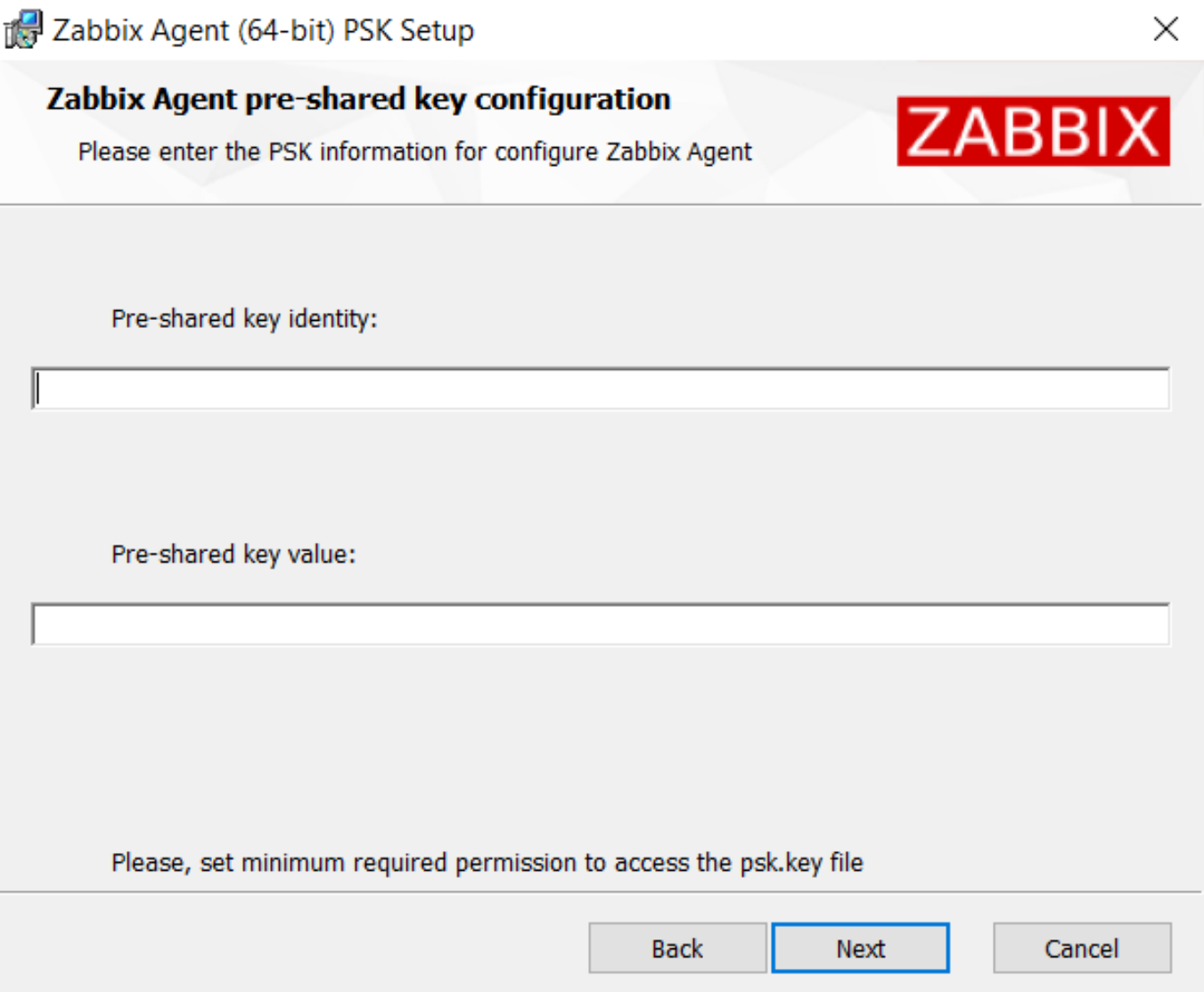


Aceite a licença para prosseguir para a próxima etapa.



Especifique os seguintes parâmetros.

Parâmetros	Descrição
Nome do Host	Especifique o host name.
IP/DNS do servidor Zabbix	Especifique o IP/DNS do servidor Zabbix.
Porta de comunicação do agente	Especifique a porta de comunicação (10050 por padrão).
Servidor ou Proxy para ativar verificação	Especifique o IP/DNS do servidor/proxy Zabbix para verificação ativa do agente.
Habilitar PSK	Marque como selecionado para habilitar suporte TLS via chaves pré-compartilhadas.
Adicione a localização do agente PATH	Adicione a localização do agente à variável PATH.



Zabbix Agent (64-bit) PSK Setup

Zabbix Agent pre-shared key configuration

Please enter the PSK information for configure Zabbix Agent

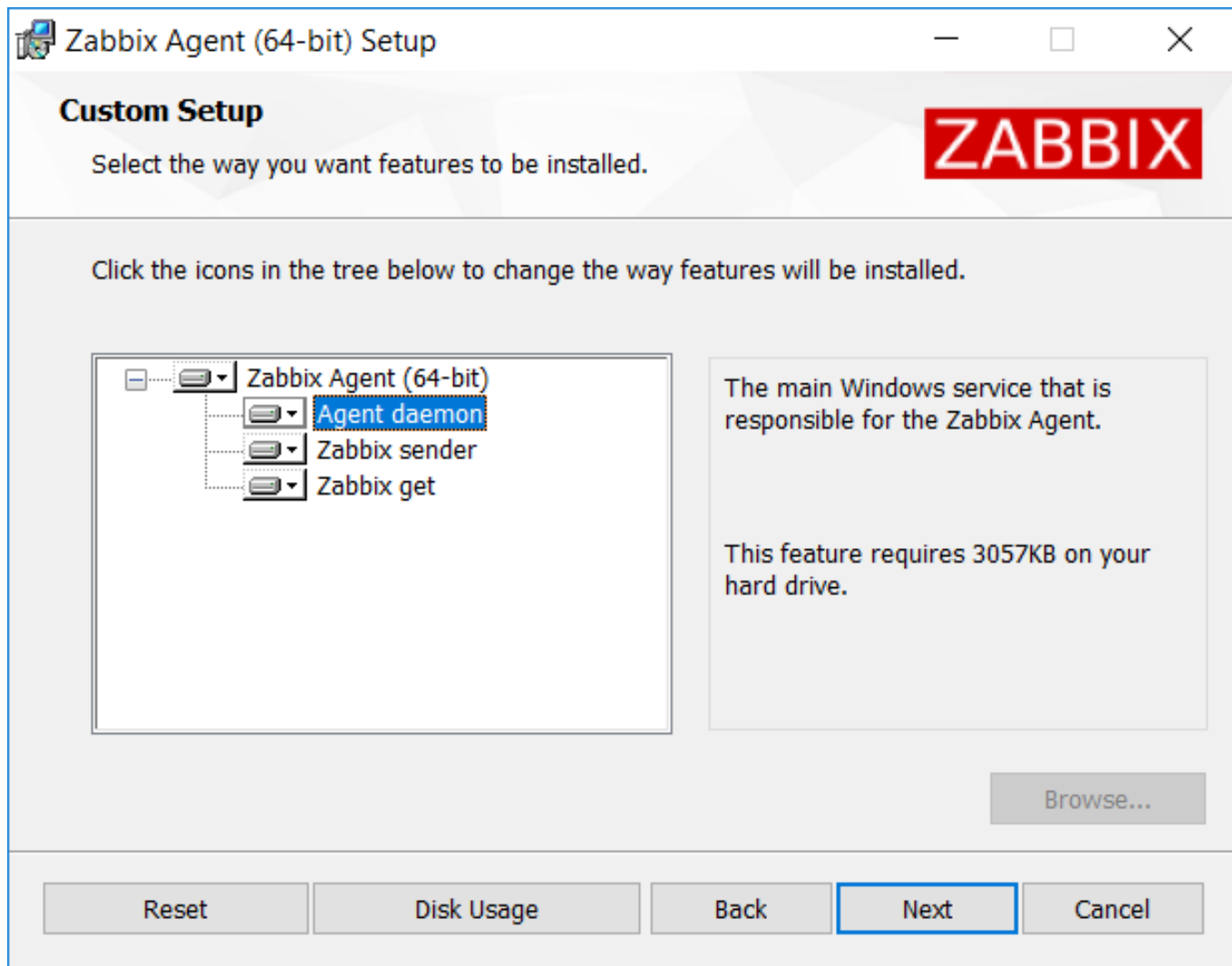
Pre-shared key identity:

Pre-shared key value:

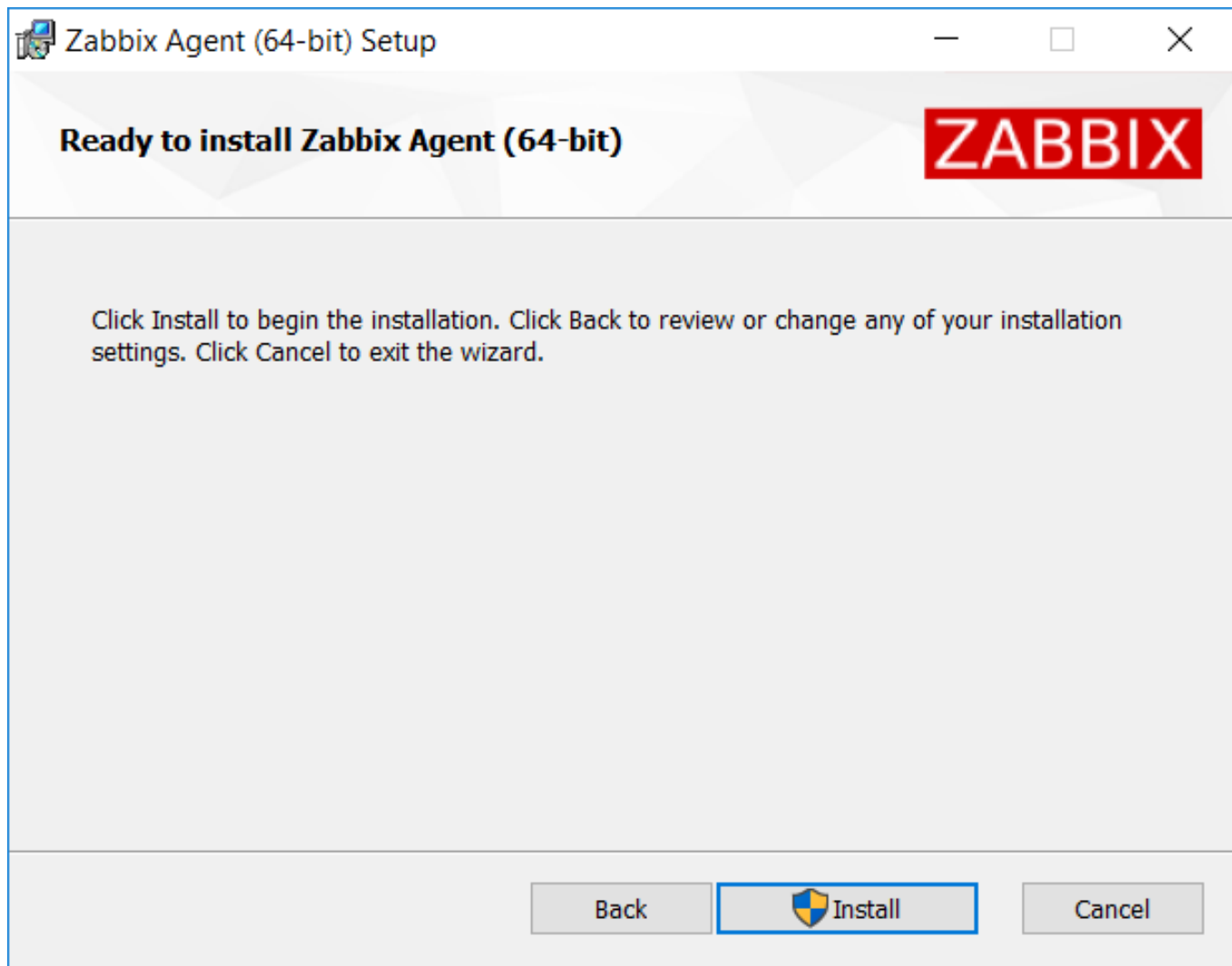
Please, set minimum required permission to access the psk.key file

Back Next Cancel

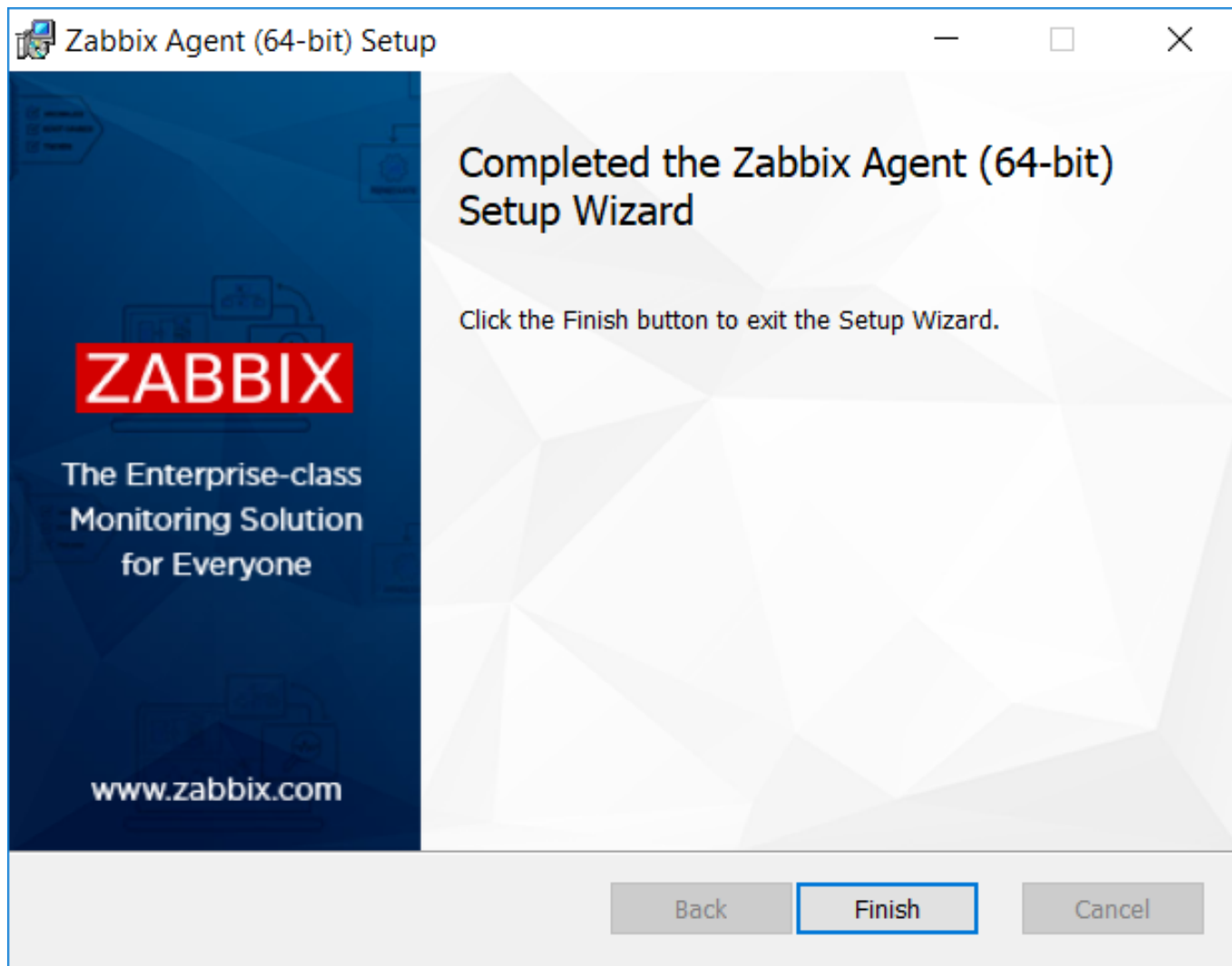
Insira a identidade e o valor da chave pré-compartilhada. Esta etapa só está disponível se você marcou Habilitar PSK na etapa anterior..



Selecione os componentes Zabbix para instalação - **Zabbix agent daemon**, **Zabbix sender**, **Zabbix get**.



Os componentes do Zabbix junto com o arquivo de configuração serão instalados em uma pasta * Zabbix Agent * em Arquivos de Programas. zabbix _agentd.exe será configurado como serviço do Windows com inicialização automática.



Instalação baseada em linha de comandos

Parâmetros suportados

O seguinte conjunto de parâmetros é suportado pelos MSIs criados:

Número	Parâmetro	Descrição
1	LOGTYPE	
2	LOGFILE	
3	SERVER	
4	LISTENPORT	
5	SERVERACTIVE	
6	HOSTNAME	
7	TIMEOUT	
8	TLSCONNECT	
9	TLSACCEPT	
10	TLSPSKIDENTITY	
11	TLSPSKFILE	
12	TLSPSKVALUE	
13	TLSCAFILE	
14	TLSCRLFILE	
15	TLSSERVERCERTISSUER	
16	TLSSERVERCERTSUBJECT	
17	TLSCERTFILE	
18	TLSKEYFILE	
19	LISTENIP	
20	HOSTINTERFACE	
21	HOSTMETADATA	
22	HOSTMETADATAITEM	

Número	Parâmetro	Descrição
23	STATUSPORT	Somente para Zabbix agent 2.
24	ENABLEPERSISTENTBUFFER	Somente para Zabbix agent 2.
25	PERSISTENTBUFFERPERIOD	Somente para Zabbix agent 2.
26	PERSISTENTBUFFERFILE	Somente para Zabbix agent 2.
27	INSTALLFOLDER	
28	ENABLEPATH	
29	SKIP	SKIP=fw - não instalar regra de exceção de firewall
30	INCLUDE	Sequência de inclusões separadas por ;
31	ALLOWDENYKEY	Sequência de "AllowKey" e "DenyKey" parameters separados por ;. Use \\; para escapar o delimitador. Exemplo: ALLOWDENYKEY="AllowKey=system.run[type c:\windows\system32\drivers\etc\hosts];DenyKey=system.run[*]"
32	ADDPROGRAM	Uma lista separada por vírgulas de programas para instalar. Valores possíveis: AgentProgram, GetProgram, SenderProgram Por exemplo, ADDPROGRAM=AgentProgram,GetProgram
33	ADDLOCAL	Uma lista separada por vírgulas de programas para instalar. Valores possíveis: AgentProgram, GetProgram, SenderProgram Por exemplo, ADDLOCAL=AgentProgram,SenderProgram
34	CONF	Especifique o caminho para o arquivo de configuração personalizado, por exemplo, CONF=c:\full\path\to\user.conf

Para instalar, você pode executar, por exemplo:

```
SET INSTALLFOLDER=C:\Program Files\za
```

```
msiexec /l*v log.txt /i zabbix_agent-6.0.0-x86.msi /qn^
LOGTYPE=file^
LOGFILE="%INSTALLFOLDER%\za.log"^
SERVER=192.168.6.76^
LISTENPORT=12345^
SERVERACTIVE=:1^
HOSTNAME=myHost^
TLSCONNECT=psk^
TLSACCEPT=psk^
TLSPSKIDENTITY=MyPSKID^
TLSPSKFILE="%INSTALLFOLDER%\mykey.psk"^
TLSCAFILE="c:\temp\f.txt1"^
TLSCRLFILE="c:\temp\f.txt2"^
TLSSERVERCERTISSUER="My CA"^
TLSSERVERCERTSUBJECT="My Cert"^
TLSCERTFILE="c:\temp\f.txt5"^
TLSKEYFILE="c:\temp\f.txt6"^
ENABLEPATH=1^
INSTALLFOLDER="%INSTALLFOLDER%"^
SKIP=fw^
ALLOWDENYKEY="DenyKey=vfs.file.contents[/etc/passwd]"
```

ou

```
msiexec /l*v log.txt /i zabbix_agent-6.0.0-x86.msi /qn^
SERVER=192.168.6.76^
TLSCONNECT=psk^
TLSACCEPT=psk^
TLSPSKIDENTITY=MyPSKID^
TLSPSKVALUE=1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952
```

Se tanto TLSPSKFILE quanto TLSPSKVALUE forem passados, então TLSPSKVALUE será gravado em TLSPSKFILE.

5 Instalação de agente Mac OS pelo PKG

Visão geral

O agente Zabbix para Mac OS pode ser instalado pelos pacotes de instalação PKG disponíveis para [download](#). Versões com ou sem criptografia estão disponíveis.

Instalando agente

O agente pode ser instalado usando a interface gráfica ou pela linha de comando, por exemplo:

```
sudo installer -pkg zabbix_agent-5.4.0-macos-amd64-openssl.pkg -target /
```

Certifique-se de utilizar a versão correta do pacote Zabbix no comando. Esta deve corresponder ao nome do pacote baixado.

Executando o agente

O agente iniciará automaticamente após a instalação ou reinício.

Você pode editar o arquivo de configuração em `/usr/local/etc/zabbix/zabbix_agentd.conf` se necessário.

Para iniciar o agente manualmente, você pode executar:

```
sudo launchctl start com.zabbix.zabbix_agentd
```

Para parar o agente manualmente:

```
sudo launchctl stop com.zabbix.zabbix_agentd
```

Durante a atualização, o arquivo de configuração existente não é sobrescrito. Em vez disso um novo arquivo `zabbix_agentd.conf.NEW` é criado para ser usado na revisão e atualização das configurações existentes, se necessário. Lembre-se de reiniciar o agente após alterações manuais nos arquivos de configuração.

Correção e remoção do agente

Esta seção lista alguns comandos úteis que podem ser usados para solução de problemas e remoção da instalação do Zabbix Agent.

Veja se o Zabbix Agent está sendo executado:

```
ps aux | grep zabbix_agentd
```

Veja se o Zabbix Agent foi instalado pelos pacotes:

```
$ pkgutil --pkgs | grep zabbix
com.zabbix.pkg.ZabbixAgent
```

Confira os arquivos que foram instalados pelo pacote de instalação (note que a barra inicial / não é mostrada nesta visualização):

```
$ pkgutil --only-files --files com.zabbix.pkg.ZabbixAgent
Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
usr/local/bin/zabbix_get
usr/local/bin/zabbix_sender
usr/local/etc/zabbix/zabbix_agentd/userparameter_examples.conf.NEW
usr/local/etc/zabbix/zabbix_agentd/userparameter_mysql.conf.NEW
usr/local/etc/zabbix/zabbix_agentd.conf.NEW
usr/local/sbin/zabbix_agentd
```

Pare o Zabbix Agent se ele foi iniciado com o `launchctl`:

```
sudo launchctl unload /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
```

Remova os arquivos (incluindo configurações e logs) que foram instalados com o pacote de instalação:

```
sudo rm -f /Library/LaunchDaemons/com.zabbix.zabbix_agentd.plist
sudo rm -f /usr/local/sbin/zabbix_agentd
sudo rm -f /usr/local/bin/zabbix_get
sudo rm -f /usr/local/bin/zabbix_sender
sudo rm -rf /usr/local/etc/zabbix
sudo rm -rf /var/log/zabbix
```

Utilize o comando `forget` para o Zabbix Agent:

```
sudo pkgutil --forget com.zabbix.pkg.ZabbixAgent
```

6 Lançamentos instáveis

Overview

Packages for minor Zabbix version (i.e. Zabbix 6.0.x) release candidates are provided starting with Zabbix 6.0.9.

The instructions below are for enabling unstable Zabbix release repositories (disabled by default).

First, install or update to the latest zabbix-release package. To enable rc packages on your system do the following:

Red Hat Enterprise Linux

Open the `/etc/yum.repos.d/zabbix.repo` file and set `enabled=1` for the `zabbix-unstable` repo.

```
[zabbix-unstable]
name=Zabbix Official Repository (unstable) - $basearch
baseurl=https://repo.zabbix.com/zabbix/5.5/rhel/8/$basearch/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
```

Debian/Ubuntu

Open the `/etc/apt/sources.list.d/zabbix.list` and uncomment "Zabbix unstable repository".

```
#### Zabbix unstable repository
deb https://repo.zabbix.com/zabbix/5.5/debian bullseye main
deb-src https://repo.zabbix.com/zabbix/5.5/debian bullseye main
```

SUSE

Open the `/etc/zypp/repos.d/zabbix.repo` file and set `enable=1` for the `zabbix-unstable` repo.

```
[zabbix-unstable]
name=Zabbix Official Repository
type=rpm-md
baseurl=https://repo.zabbix.com/zabbix/5.5/sles/15/x86_64/
gpgcheck=1
gpgkey=https://repo.zabbix.com/zabbix/5.5/sles/15/x86_64/repoata/repomd.xml.key
enabled=1
update=1
```

5 Instalação a partir de containers

Docker O Zabbix fornece [Docker](#) imagens para cada componente do Zabbix como containers portáteis e autosuficientes, acelerando o procedimento de implantação e atualização.

Os componentes do Zabbix vêm com suporte para banco de dados MySQL e PostgreSQL, e suporte para servidores web Apache2 e Nginx. Essas imagens são divididas em diferentes imagens.

Attention:

Desde a versão do Zabbix 6.0, os triggers determinísticos precisam ser criados durante a instalação. Se o log binário estiver habilitado para MySQL/MariaDB, isso requer privilégios de superuser ou a configuração da variável/configuration parameter `log_bin_trust_function_creators = 1`. Para instruções sobre como configurar a variável, consulte [Database creation scripts](#). Observe que, se estiver executando a partir de um console, a variável será definida apenas temporariamente e será removida quando um Docker for reiniciado. Nesse caso, mantenha o serviço SQL rodando, apenas pare o serviço `zabbix-server` executando `'docker compose down zabbix-server'` e, em seguida, `'docker compose up -d zabbix-server'`. Se preferir, você pode definir essa variável no arquivo de configuração.

Imagens base Docker

Os componentes Zabbix são disponibilizados em imagens construídas sobre Ubuntu, Alpine Linux e CentOS:

Image	Version
alpine	3.12
ubuntu	20.04 (focal)
centos	8

Todas as imagens são configuradas para reconstruir as últimas imagens caso alguma das imagens base receba atualização.

Arquivos fonte para Docker

Todos podem acompanhar alterações nos arquivos Docker através do [repositório oficial](#) no [github.com](#). Você também pode realizar um fork do projeto ou criar suas próprias imagens baseadas nos arquivos Docker oficiais.

Estrutura

Todos os componentes Zabbix estão disponíveis nos seguintes repositórios Docker:

- Zabbix Agent - [zabbix/zabbix-agent](#)
- Zabbix Server
 - Zabbix Server com suporte a MySQL - [zabbix/zabbix-server-mysql](#)
 - Zabbix Server com suporte a PostgreSQL - [zabbix/zabbix-server-pgsql](#)
- Zabbix Web-Interface
 - Zabbix Web-Interface baseado em Apache2 Web Server com MySQL - [zabbix/zabbix-web-apache-mysql](#)
 - Zabbix Web-Interface baseado em Apache2 Web Server com PostgreSQL - [zabbix/zabbix-web-apache-pgsql](#)
 - Zabbix Web-Interface baseado em Nginx Web Server com MySQL - [zabbix/zabbix-web-nginx-mysql](#)
 - Zabbix Web-Interface baseado em Nginx Web Server com PostgreSQL - [zabbix/zabbix-web-nginx-pgsql](#)
- Zabbix Proxy
 - Zabbix Proxy com suporte a SQLite3 - [zabbix/zabbix-proxy-sqlite3](#)
 - Zabbix Proxy com suporte a MySQL - [zabbix/zabbix-proxy-mysql](#)
- Zabbix Java Gateway - [zabbix/zabbix-java-gateway](#)

Adicionalmente há suporte para SNMP Trap. Ele é fornecido através de repositório adicional ([zabbix/zabbix-snmptraps](#)) baseado apenas em Ubuntu Trusty. Tal solução pode ser associada com Zabbix Server e Zabbix Proxy.

Versões

Cada repositório de componentes Zabbix contém as seguintes tags:

- latest - versão estável mais recente de um componente Zabbix baseado em imagem Alpine Linux
- alpine-latest - versão estável mais recente de um componente Zabbix baseado em imagem Alpine Linux
- ubuntu-latest - versão estável mais recente de um componente Zabbix baseado em imagem Ubuntu
- alpine-5.4-latest - versão secundária (minor version) mais recente de um componente Zabbix 5.4 baseado em imagem Alpine Linux
- ubuntu-5.4-latest - versão secundária (minor version) mais recente de um componente Zabbix 5.4 baseado em imagem Ubuntu
- alpine-5.4.* - diferentes versões secundárias (minor versions) de um componente Zabbix 5.4 baseado em imagem Alpine Linux, onde * representa a versão secundária do componente Zabbix
- ubuntu-5.4.* - diferentes versões secundárias (minor versions) de um componente Zabbix 5.4 baseado em imagem Ubuntu, onde * representa a versão secundária do componente Zabbix

Uso

Variáveis de ambiente

Todas as imagens de componentes Zabbix fornecem variáveis de ambiente para controlar a configuração. Estas variáveis de ambiente estão listadas em cada repositório dos componentes. Tais variáveis são opções vindas dos arquivos de configuração do Zabbix, mas com método de nomenclatura diferenciado. Por exemplo, ZBX_LOGSLOWQUERIES é o mesmo que a opção LogSlowQueries dos arquivos de configuração do Zabbix Server e Zabbix Proxy.

Attention:

Algumas opções de configuração não podem ser alteradas. Por exemplo, PIDFile and LogType.

Alguns dos componentes possuem variáveis de ambiente específicas, que não existem nos arquivos de configuração oficiais do Zabbix:

Variável	Componentes	Descrição
DB_SERVER_HOST	Server Proxy Web interface	Esta variável representa o IP ou nome DNS do servidor MySQL ou PostgreSQL. Por padrão, seu valor é mysql-server ou postgres-server para MySQL ou PostgreSQL, respectivamente.
DB_SERVER_PORT	Server Proxy Web interface	Esta variável representa a porta do servidor MySQL ou PostgreSQL. Por padrão, seu valor é '3306' ou '5432', respectivamente.

MYSQL_USER	Server Proxy Web-interface	Usuário do banco de dados MySQL. Por padrão, seu valor é 'zabbix'.
MYSQL_PASSWORD	Server Proxy Web interface	Senha do banco de dados MySQL. Por padrão, seu valor é 'zabbix'.
MYSQL_DATABASE	Server Proxy Web interface	Nome do banco de dados do Zabbix. Por padrão, seu valor é 'zabbix' para Zabbix Server e 'zabbix_proxy' para Zabbix Proxy.
POSTGRES_USER	Server Web interface	Usuário do banco de dados PostgreSQL. Por padrão, seu valor é 'zabbix'.
POSTGRES_PASSWORD	Server Web interface	Senha do banco de dados PostgreSQL. Por padrão, seu valor é 'zabbix'.
POSTGRES_DB	Server Web interface	Nome do banco de dados do Zabbix. Por padrão, seu valor é 'zabbix' para Zabbix Server e 'zabbix_proxy' para Zabbix Proxy.
PHP_TZ	Web-interface	Timezone no formato PHP. Lista completa de timezones suportados disponível em php.net . Por padrão, seu valor é 'Europe/Riga'.
ZBX_SERVER_NAME	Web interface	Nome de instalação visível no canto superior direito da interface web. Por padrão, seu valor é 'Zabbix Docker'
ZBX_JAVAGATEWAY_ENABLE	Server Proxy	Habilita comunicação com Zabbix Java Gateway para coleta de verificações relacionadas ao Java. Por padrão, seu valor é "false"
ZBX_ENABLE_SNMP_TRAPS	Server Proxy	Habilita a funcionalidade SNMP Trap. Requer instância zabbix-snmptraps e volume /var/lib/zabbix/snmptraps compartilhado para Zabbix Server ou Zabbix Proxy.

Volumes

As imagens permitem utilizar alguns pontos de montagem. Estes pontos de montagem são diferentes e dependem do tipo de componente Zabbix:

Volume	Descrição
Zabbix Agent	
/etc/zabbix/zabbix_agentd.d	O volume permite incluir arquivos *.conf e estende o Zabbix Agent usando a funcionalidade <code>UserParameter</code>
/var/lib/zabbix/modules	O volume permite carregar módulos adicionais e estende o Zabbix Agent usando a funcionalidade <code>LoadModule</code>
/var/lib/zabbix/enc	O volume é usado para armazenar arquivos relacionados a TLS. Estes nomes de arquivo são especificados usando as variáveis de ambiente ZBX_TLSCAFILE, ZBX_TLSCRLFILE, ZBX_TLSKEY_FILE e ZBX_TLSPSKFILE
Zabbix Server	
/usr/lib/zabbix/alertscripts	O volume é usado para scripts de alerta customizados. É o parâmetro <code>AlertScriptsPath</code> no zabbix_server.conf
/usr/lib/zabbix/externalscripts	O volume é usado por verificações externas . É o parâmetro <code>ExternalScripts</code> no zabbix_server.conf
/var/lib/zabbix/modules	O volume permite carregar módulos adicionais e estende o Zabbix Server usando a funcionalidade <code>LoadModule</code>

/var/lib/zabbix/enc	O volume é usado para armazenar arquivos relacionados a TLS. Estes nomes de arquivo são especificados usando as variáveis de ambiente ZBX_TLSCAFILE, ZBX_TLSCRLFILE, ZBX_TLSKEY_FILE e ZBX_TLSPSKFILE
/var/lib/zabbix/ssl/certs	O volume é usado como local para os arquivos de certificado de cliente SSL para autenticação de cliente. É o parâmetro SSLCertLocation no zabbix_server.conf
/var/lib/zabbix/ssl/keys	O volume é usado como local para os arquivos de chave privada SSL para autenticação de cliente. É o parâmetro SSLKeyLocation no zabbix_server.conf
/var/lib/zabbix/ssl/ssl_ca	O volume é usado como local para arquivos de certificado de autoridade (CA) para verificação SSL de certificado de servidor. É o parâmetro SSLCALocation no zabbix_server.conf
/var/lib/zabbix/snmptraps	O volume é usado como local para o arquivo snmptraps.log. Ele pode ser compartilhado pelo container zabbix-snmptraps e herdado usando a opção volumes_from do Docker durante a criação de uma nova instância do Zabbix Server. A funcionalidade de processamento de SNMP Trap pode ser habilitada pelo uso do volume compartilhado e alterando a variável de ambiente ZBX_ENABLE_SNMP_TRAPS para 'true'
/var/lib/zabbix/mibs	O volume permite adicionar novos arquivos MIB. Ele não suporta subdiretórios, assim todas as MIBs devem estar em /var/lib/zabbix/mibs
Zabbix Proxy	
/usr/lib/zabbix/externalscripts	O volume é usado por verificações externas. É o parâmetro ExternalScripts no zabbix_proxy.conf
/var/lib/zabbix/modules	O volume permite carregar módulos adicionais e estender o Zabbix Proxy usando a funcionalidade LoadModule
/var/lib/zabbix/enc	O volume é usado para armazenar arquivos relacionados a TLS. Estes nomes de arquivo são especificados usando as variáveis de ambiente ZBX_TLSCAFILE, ZBX_TLSCRLFILE, ZBX_TLSKEY_FILE e ZBX_TLSPSKFILE
/var/lib/zabbix/ssl/certs	O volume é usado como local para os arquivos de certificado de cliente SSL para autenticação de cliente. É o parâmetro SSLCertLocation no zabbix_proxy.conf
/var/lib/zabbix/ssl/keys	O volume é usado como local para os arquivos de chave privada SSL para autenticação de cliente. É o parâmetro SSLKeyLocation no zabbix_proxy.conf
/var/lib/zabbix/ssl/ssl_ca	O volume é usado como local para arquivos de certificado de autoridade (CA) para verificação SSL de certificado de servidor. É o parâmetro SSLCALocation no zabbix_proxy.conf
/var/lib/zabbix/snmptraps	O volume é usado como local para o arquivo snmptraps.log. Ele pode ser compartilhado pelo container zabbix-snmptraps e herdado usando a opção volumes_from do Docker durante a criação de uma nova instância do Zabbix Server. A funcionalidade de processamento de SNMP Trap pode ser habilitada pelo uso do volume compartilhado e alterando a variável de ambiente ZBX_ENABLE_SNMP_TRAPS para 'true'
/var/lib/zabbix/mibs	O volume permite adicionar novos arquivos MIB. Ele não suporta subdiretórios, assim todas as MIBs devem estar em /var/lib/zabbix/mibs
Zabbix Web Interface baseada em Apache2 Web Server	
/etc/ssl/apache2	O volume permite habilitar o protocolo HTTPS para o Zabbix Web Interface. O volume deve conter os dois arquivos ssl.crt e ssl.key preparados para conexões Apache2 SSL
Zabbix Web Interface baseada em Nginx Web Server	

<code>/etc/ssl/nginx</code>	O volume permite habilitar o protocolo HTTPS para o Zabbix Web Interface. O volume deve conter os arquivos <code>ssl.crt</code> , <code>ssl.key</code> e <code>dhparam.pem</code> preparados para conexões Nginx SSL
Zabbix snmptraps <code>/var/lib/zabbix/snmptraps</code>	O volume contém o arquivo de log <code>snmptraps.log</code> nomeado com os SNMP Traps recebidos
<code>/var/lib/zabbix/mibs</code>	O volume permite adicionar novos arquivos MIB. Ele não suporta subdiretórios, assim todas as MIBs devem estar em <code>/var/lib/zabbix/mibs</code>

Para informação adicional utilize os repositórios oficiais do Zabbix no Docker Hub.

Exemplos de uso

Exemplo 1

O exemplo demonstra como executar o Zabbix Server com suporte a banco de dados MySQL, Zabbix Web Interface baseada em Nginx Web Server e Zabbix Java Gateway.

1. Crie uma rede dedicada aos contêineres dos componentes Zabbix:

```
# docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Inicie uma instância vazia de MySQL Server

```
# docker run --name mysql-server -t \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e MYSQL_ROOT_PASSWORD="root_pwd" \
  --network=zabbix-net \
  -d mysql:8.0 \
  --restart unless-stopped \
  --character-set-server=utf8 --collation-server=utf8_bin \
  --default-authentication-plugin=mysql_native_password
```

3. Inicie uma instância do Zabbix Java Gateway

```
# docker run --name zabbix-java-gateway -t \
  --network=zabbix-net \
  --restart unless-stopped \
  -d zabbix/zabbix-java-gateway:alpine-5.4-latest
```

4. Inicie a instância do Zabbix Server e a associe à instância do MySQL Server recém-criada

```
# docker run --name zabbix-server-mysql -t \
  -e DB_SERVER_HOST="mysql-server" \
  -e MYSQL_DATABASE="zabbix" \
  -e MYSQL_USER="zabbix" \
  -e MYSQL_PASSWORD="zabbix_pwd" \
  -e MYSQL_ROOT_PASSWORD="root_pwd" \
  -e ZBX_JAVAGATEWAY="zabbix-java-gateway" \
  --network=zabbix-net \
  -p 10051:10051 \
  --restart unless-stopped \
  -d zabbix/zabbix-server-mysql:alpine-5.4-latest
```

Note:

A instância do Zabbix Server expõe a porta 10051/TCP (Zabbix trapper) para máquina local.

5. Inicie o Zabbix Web Interface e o associe às instâncias do MySQL Server e Zabbix Server

```
# docker run --name zabbix-web-nginx-mysql -t \
  -e ZBX_SERVER_HOST="zabbix-server-mysql" \
  -e DB_SERVER_HOST="mysql-server" \
  -e MYSQL_DATABASE="zabbix" \
```



```
-e MYSQL_USER="zabbix" \
-e MYSQL_PASSWORD="zabbix_pwd" \
-e MYSQL_ROOT_PASSWORD="root_pwd" \
--network=zabbix-net \
-p 80:8080 \
--restart unless-stopped \
-d zabbix/zabbix-web-nginx-mysql:alpine-5.4-latest
```

Note:

A instância do Zabbix Web Interface expõe a porta 80/TCP (HTTP) para a máquina local.

Exemplo 2

O exemplo demonstra como executar o Zabbix Server com suporte a PostgreSQL, Zabbix Web Interface baseada em Nginx Web Server e funcionalidade SNMP Trap.

1. Crie uma rede dedicada para os contêineres dos componentes Zabbix:

```
# docker network create --subnet 172.20.0.0/16 --ip-range 172.20.240.0/20 zabbix-net
```

2. Inicie uma instância vazia do PostgreSQL Server

```
# docker run --name postgres-server -t \
-e POSTGRES_USER="zabbix" \
-e POSTGRES_PASSWORD="zabbix_pwd" \
-e POSTGRES_DB="zabbix" \
--network=zabbix-net \
--restart unless-stopped \
-d postgres:latest
```

3. Inicie uma instância do Zabbix snmptraps

```
# docker run --name zabbix-snmptraps -t \
-v /zbx_instance/snmptraps:/var/lib/zabbix/snmptraps:rw \
-v /var/lib/zabbix/mibs:/usr/share/snmp/mibs:ro \
--network=zabbix-net \
-p 162:1162/udp \
--restart unless-stopped \
-d zabbix/zabbix-snmptraps:alpine-5.4-latest
```

Note:

A instância do Zabbix snmptrap expõe a porta 162/UDP (SNMP traps) para a máquina local.

4. Inicie a instância do Zabbix Server e a associe com a instância do PostgreSQL Server

```
# docker run --name zabbix-server-pgsql -t \
-e DB_SERVER_HOST="postgres-server" \
-e POSTGRES_USER="zabbix" \
-e POSTGRES_PASSWORD="zabbix_pwd" \
-e POSTGRES_DB="zabbix" \
-e ZBX_ENABLE_SNMP_TRAPS="true" \
--network=zabbix-net \
-p 10051:10051 \
--volumes-from zabbix-snmptraps \
--restart unless-stopped \
-d zabbix/zabbix-server-pgsql:alpine-5.4-latest
```

Note:

A instância do Zabbix Server expõe a porta 10051/TCP (Zabbix trapper) para a máquina local.

5. Inicie o Zabbix Web Interface e o associe às instâncias do PostgreSQL Server e Zabbix Server

```
# docker run --name zabbix-web-nginx-pgsql -t \
-e ZBX_SERVER_HOST="zabbix-server-pgsql" \
-e DB_SERVER_HOST="postgres-server" \
-e POSTGRES_USER="zabbix" \
```

```
-e POSTGRES_PASSWORD="zabbix_pwd" \
-e POSTGRES_DB="zabbix" \
--network=zabbix-net \
-p 443:8443 \
-p 80:8080 \
-v /etc/ssl/nginx:/etc/ssl/nginx:ro \
--restart unless-stopped \
-d zabbix/zabbix-web-nginx-pgsql:alpine-5.4-latest
```

Note:

A instância do Zabbix Web Interface expõe a porta 443/TCP (HTTPS) para a máquina local.
O diretório /etc/ssl/nginx deve conter certificado com o nome requerido.

Exemplo 3

O exemplo demonstra como executar o Zabbix Server com suporte a MySQL, Zabbix Web Interface baseada em Nginx Web Server e Zabbix Java Gateway usando podman em Red Hat 8.

1. Crie um novo pod com o nome zabbix e exponha as portas (web-interface, Zabbix server trapper):

```
podman pod create --name zabbix -p 80:8080 -p 10051:10051
```

2. (optional) Inicie o contêiner do Zabbix Agent no pod zabbix:

```
podman run --name zabbix-agent \
-e ZBX_SERVER_HOST="127.0.0.1,localhost" \
--restart=always \
--pod=zabbix \
-d registry.connect.redhat.com/zabbix/zabbix-agent-50:latest
```

3. Crie o diretório ./mysql/ na máquina e inicie o Oracle MySQL Server 8.0:

```
podman run --name mysql-server -t \
-e MYSQL_DATABASE="zabbix" \
-e MYSQL_USER="zabbix" \
-e MYSQL_PASSWORD="zabbix_pwd" \
-e MYSQL_ROOT_PASSWORD="root_pwd" \
-v ./mysql:/var/lib/mysql:Z \
--restart=always \
--pod=zabbix \
-d mysql:8.0 \
--character-set-server=utf8 --collation-server=utf8_bin \
--default-authentication-plugin=mysql_native_password
```

3. Inicie o contêiner do Zabbix Server:

```
podman run --name zabbix-server-mysql -t \
-e DB_SERVER_HOST="127.0.0.1" \
-e MYSQL_DATABASE="zabbix" \
-e MYSQL_USER="zabbix" \
-e MYSQL_PASSWORD="zabbix_pwd" \
-e MYSQL_ROOT_PASSWORD="root_pwd" \
-e ZBX_JAVAGATEWAY="127.0.0.1" \
--restart=always \
--pod=zabbix \
-d registry.connect.redhat.com/zabbix/zabbix-server-mysql-50
```

4. Inicie o contêiner do Zabbix Java Gateway:

```
podman run --name zabbix-java-gateway -t \
--restart=always \
--pod=zabbix \
-d registry.connect.redhat.com/zabbix/zabbix-java-gateway-50
```

5. Inicie o contêiner do Zabbix Web-Interface:

```
podman run --name zabbix-web-mysql -t \
-e ZBX_SERVER_HOST="127.0.0.1" \
-e DB_SERVER_HOST="127.0.0.1" \
```

```
-e MYSQL_DATABASE="zabbix" \
-e MYSQL_USER="zabbix" \
-e MYSQL_PASSWORD="zabbix_pwd" \
-e MYSQL_ROOT_PASSWORD="root_pwd" \
--restart=always \
--pod=zabbix \
-d registry.connect.redhat.com/zabbix/zabbix-web-mysql-50
```

Note:

O Pod zabbix expõe a porta 80/TCP (HTTP) para a máquina local a partir da porta 8080/TCP do contêiner zabbix-web-mysql.

Docker Compose A Zabbix também fornece arquivos compose para definição e execução multi-container dos componentes Zabbix em Docker. Estes arquivos compose estão disponíveis no repositório oficial do Zabbix no github.com: <https://github.com/zabbix/zabbix-docker>. Estes arquivos compose estão disponíveis como exemplos, e estão sobrecarregados. Por exemplo, eles contém proxies com suporte a MySQL e SQLite3.

Há algumas versões diferentes de arquivos compose:

Nome do arquivo	Descrição
docker-compose_v3_alpine_mysql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em Alpine Linux com suporte a MySQL.
docker-compose_v3_alpine_mysql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em Alpine Linux com suporte a MySQL.
docker-compose_v3_alpine_pgsql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em Alpine Linux com suporte a PostgreSQL.
docker-compose_v3_alpine_pgsql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em Alpine Linux com suporte a PostgreSQL.
docker-compose_v3_centos_mysql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em CentOS 8 com suporte a MySQL.
docker-compose_v3_centos_mysql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em CentOS 8 com suporte a MySQL.
docker-compose_v3_centos_pgsql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em CentOS 8 com suporte a PostgreSQL.
docker-compose_v3_centos_pgsql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em CentOS 8 com suporte a PostgreSQL.
docker-compose_v3_ubuntu_mysql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em Ubuntu 20.04 com suporte a MySQL.
docker-compose_v3_ubuntu_mysql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em Ubuntu 20.04 com suporte a MySQL.
docker-compose_v3_ubuntu_pgsql_latest.yaml	O arquivo compose executa a versão mais recente dos componentes do Zabbix 5.4 em Ubuntu 20.04 com suporte a PostgreSQL.
docker-compose_v3_ubuntu_pgsql_local.yaml	O arquivo compose constrói localmente a versão mais recente do Zabbix 5.4 e executa os componentes Zabbix em Ubuntu 20.04 com suporte a PostgreSQL.

Attention:

Os arquivos compose disponíveis suportam a versão 3 do Docker Compose.

Os arquivos compose são configurados para suportar armazenamento na máquina local. O Docker Compose criará um diretório `zbx_env` na pasta com o arquivo compose quando você executar os componentes do Zabbix utilizando um arquivo compose. O diretório conterá a mesma estrutura como descrita acima na seção de **Volumes** e diretório para armazenamento de banco de dados.

Há também volumes no modo somente-leitura para os arquivos `/etc/localtime` e `/etc/timezone`.

Arquivos de ambiente

No mesmo diretório com os arquivos compose no github.com você pode encontrar arquivos com variáveis de ambiente padrão para cada componente no arquivo compose. Estes arquivos de ambiente são nomeados como `.env_<type of component>`.

Exemplos

Exemplo 1

```
# git checkout 5.4
# docker-compose -f ./docker-compose_v3_alpine_mysql_latest.yaml up -d
```

O comando fará o download das imagens mais recentes do Zabbix 5.4 para cada componente Zabbix e os executará de forma individual (detach mode).

Attention:

Não esqueça de baixar os arquivos `.env_<type of component>` do repositório oficial do Zabbix no github.com com os arquivos compose.

Exemplo 2

```
# git checkout 5.4
# docker-compose -f ./docker-compose_v3_ubuntu_mysql_local.yaml up -d
```

Este comando fará o download da imagem base Ubuntu 20.04 (focal), então construirá os componentes do Zabbix 5.4 localmente e os executará de forma individual (detach mode).

6 Instalação da interface web

Esta seção provê instruções passo a passo para a instalação da interface web do Zabbix. O Zabbix Frontend é escrito em PHP, então para seu funcionamento é necessário um servidor web com suporte ao PHP.

Note:

Você pode encontrar mais sobre a configuração SSL para frontend Zabbix frontend referindo-se a essas práticas **best practices**.

Tela de boas-vindas

Abra a URL do Zabbix Frontend no navegador. Se você instalou o Zabbix a partir dos pacotes, a URL é:

- para Apache: `http://<server_ip_or_name>/zabbix`
- para Nginx: `http://<server_ip_or_name>`

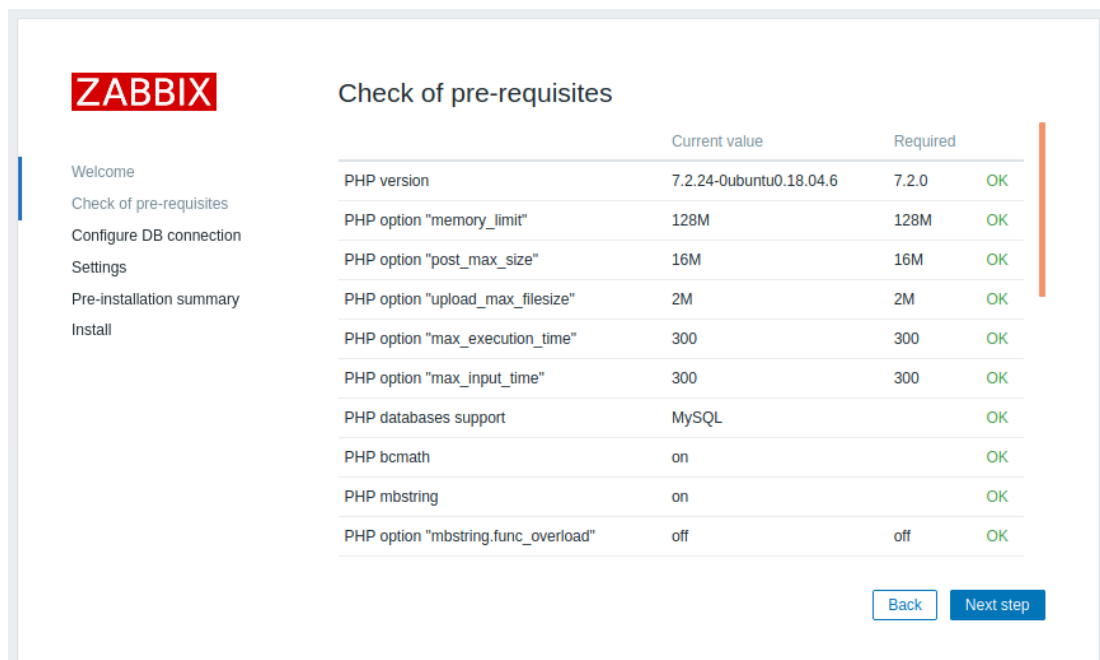
Você deverá ver a primeira tela do assistente de instalação do Frontend.

Utilize o menu Default language (idioma padrão) para alterar o idioma padrão do sistema e continuar com o processo de instalação no idioma selecionado (opcional). Para mais informações, consulte **Instalação de idiomas adicionais para o Frontend**.



Verificação de pré-requisitos

Certifique-se de que todos os pré-requisitos são atendidos.



Pré-requisito	Valor mínimo	Descrição
Versão PHP	7.2.5	
Opção PHP memory_limit	128MB	Em php.ini: memory_limit = 128M
Opção PHP post_max_size	16MB	Em php.ini: post_max_size = 16M
Opção PHP upload_max_filesize	2MB	Em php.ini: upload_max_filesize = 2M
Opção PHP max_execution_time	300 segundos (valores 0 e -1 são permitidos)	Em php.ini: max_execution_time = 300
Opção PHP max_input_time	300 segundos (valores 0 e -1 são permitidos)	Em php.ini: max_input_time = 300
Opção PHP session.auto_start	Deve estar desabilitada	Em php.ini: session.auto_start = 0

Pré-requisito	Valor mínimo	Descrição
Suporte banco de dados	Um de: MySQL, Oracle, PostgreSQL.	Um dos seguintes módulos deve estar instalado: mysql, oci8, pgsql
bcmath mbstring Opção PHP mbstring.func_overload	Deve estar desabilitada	php-bcmath php-mbstring Em php.ini: mbstring.func_overload = 0
sockets		php-net-socket. Exigido para suporte a script de usuário.
gd	2.0.28	php-gd. A extensão PHP GD deve suportar imagens PNG (--with-png-dir), imagens JPEG (--with-jpeg-dir) e FreeType 2 (--with-freetype-dir).
libxml xmlwriter xmlreader ctype session gettext	2.6.15	php-xml php-xmlwriter php-xmlreader php-ctype php-session php-gettext Desde o Zabbix 2.2.1, a extensão PHP gettext não é um requisito obrigatório para a instalação do Zabbix. Se gettext não estiver instalado, o Frontend funcionará normalmente, porém, as traduções não estarão disponíveis.

Pré-requisitos opcionais também podem estar presentes na lista. Um pré-requisito opcional com falha será mostrado em laranja e apresentado como Warning. Com falha em um pré-requisito opcional, a configuração pode continuar.

Attention:

Se houver necessidade de alterar o usuário ou grupo de usuário do Apache, as permissões para a pasta de sessão devem ser verificadas. Caso contrário, a configuração do Zabbix pode ficar impedida de continuar.

Configuração de conexão com banco de dados

Informe os dados de conexão com o banco de dados. O banco de dados do Zabbix já deve estar criado.

- Welcome
- Check of pre-requisites
- Configure DB connection
- Settings
- Pre-installation summary
- Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type

Database host

Database port
0 - use default port

Database name

Store credentials in

User

Password

Database TLS encryption

Connection will not be encrypted because it uses a socket file (on Unix) or shared memory (Windows).

Se a opção Database TLS encryption (criptografia TLS do banco de dados) estiver ativada, então campos adicionais para **configuração da conexão TLS** com o banco de dados serão mostrados no formulário (apenas para MySQL ou PostgreSQL).

Se a opção HashiCorp Vault estiver selecionada para armazenamento de credenciais, campos adicionais estarão disponíveis para especificar o endpoint da API do Vault, secret path e token de autenticação:

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type:

Database host:

Database port: 0 - use default port

Database name:

Store credentials in:

Vault API endpoint:

Vault secret path:

Vault authentication token:

Database TLS encryption: ☐

[Back](#) [Next step](#)

Configurações

Informar um nome para o Zabbix Server é opcional, no entanto, caso informado, ele será mostrado na barra de menu e títulos de página.

Configure o **time zone** padrão e o tema do Frontend.

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Settings

Zabbix server name:

Default time zone:

Default theme:

[Back](#) [Next step](#)

Resumo pré-instalação

Revise o resumo das configurações antes da instalação efetiva.

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Settings

Pre-installation summary

Install

Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Database type	MySQL
Database server	localhost
Database port	default
Database name	zabbix
Database user	zabbix
Database password	*****
TLS encryption	false
Zabbix server	localhost
Zabbix server port	10051
Zabbix server name	

Back

Next step

Instalação

Se instalando o Zabbix a partir dos fontes, baixe o arquivo de configuração e salve-o dentro de conf/ no subdiretório de documentos HTML do Web Server onde você colocou os arquivos PHP do Zabbix.

ZABBIX

Welcome

Check of pre-requisites


Configure DB connection

Settings

Pre-installation summary

Install

Install



Details ▲

Cannot create the configuration file.

Unable to create the configuration file.

Alternatively, you can install it manually:


1. [Download the configuration file](#)
2. Save it as "/var/www/html/zabbix/conf/zabbix.conf.php"

Back

Finish

Opening zabbix.conf.php

You have chosen to open:



zabbix.conf.php

which is: PHP script (418 bytes)

from: http://192.168.3.194

What should Firefox do with this file?

☐ Open with

gedit (default)

☒ Save File

☐ Do this automatically for files like this from now on.

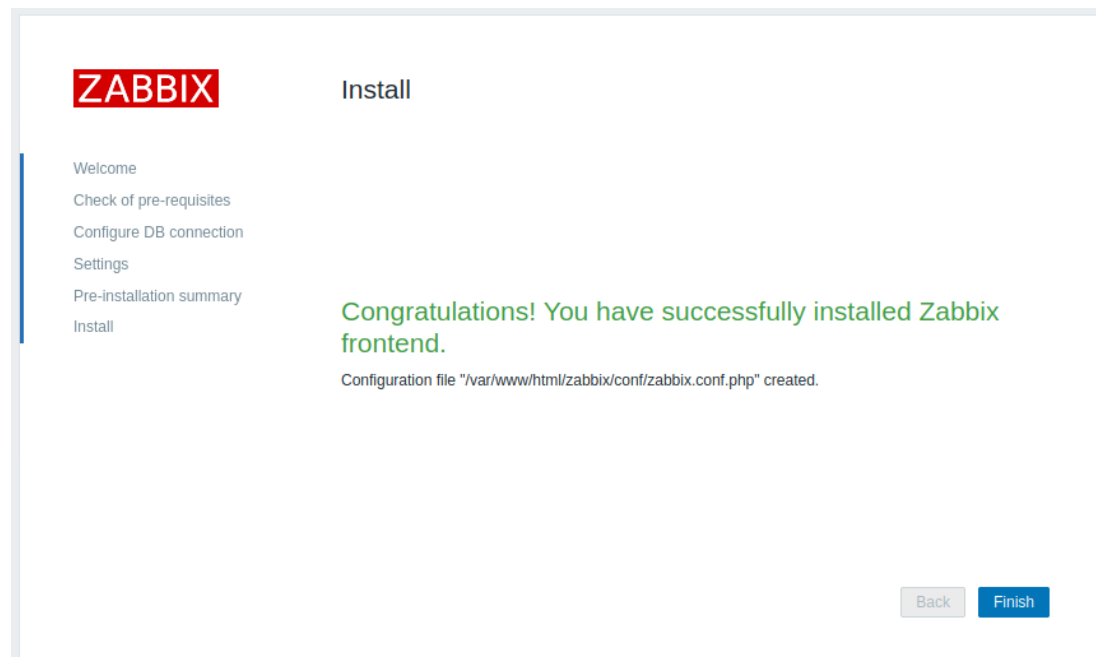
Cancel

OK

Note:

Garantindo que o usuário do Web Server tenha acesso de escrita no diretório conf/ o arquivo de configuração será salvo automaticamente e será possível continuar para o próximo passo imediatamente.

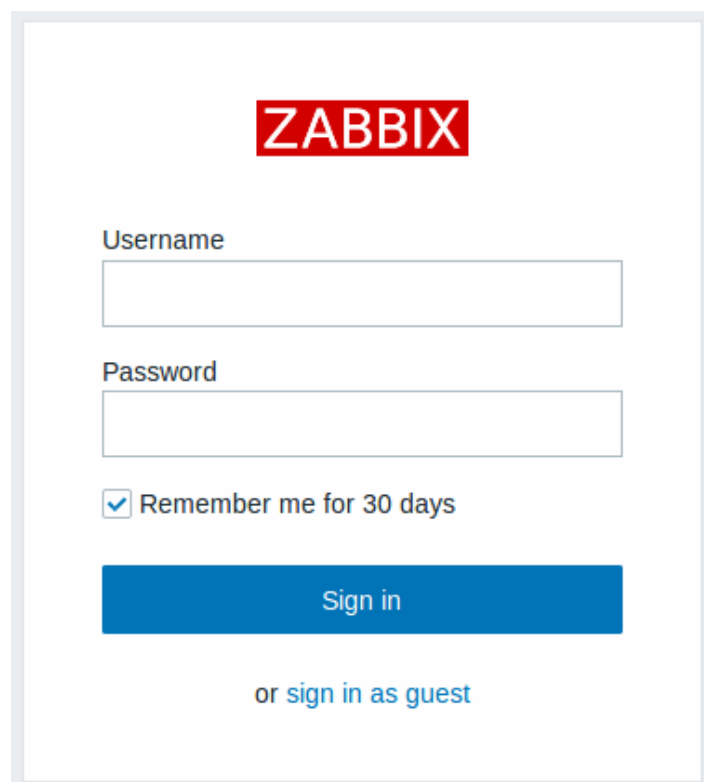
Finalize a instalação.



The image shows the Zabbix installation completion screen. At the top left is the ZABBIX logo. To its right is the word "Install". Below the logo is a vertical list of installation steps: Welcome, Check of pre-requisites, Configure DB connection, Settings, Pre-installation summary, and Install. The "Install" step is highlighted. The main content area displays a green message: "Congratulations! You have successfully installed Zabbix frontend." Below this, it states: "Configuration file "/var/www/html/zabbix/conf/zabbix.conf.php" created." At the bottom right, there are two buttons: "Back" and "Finish".

Log in

O Zabbix Frontend está pronto! O nome de usuário padrão é **Admin**, e senha **zabbix**.



The image shows the Zabbix login screen. At the top center is the ZABBIX logo. Below it are two input fields: "Username" and "Password". Under the password field is a checkbox labeled "Remember me for 30 days". Below these is a large blue "Sign in" button. At the bottom, there is a link that says "or sign in as guest".

Agora é só **começar a usar o Zabbix**.

Instalação do Frontend em Debian/Ubuntu

Visão geral

A partir da versão 5.0, o Zabbix Frontend requer PHP 7.2 ou a versão mais recente. Infelizmente, versões mais antigas do Debian e Ubuntu proveem apenas versões de PHP abaixo de 7.2.

Versões de PHP suportadas por distribuição

Distribuição	Versão PHP
Debian 10 (buster)	7.3
Debian 9 (stretch)	7.0
Debian 8 (jessie)	5.6
Ubuntu 20.04 (focal)	7.4
Ubuntu 18.04 (bionic)	7.2
Ubuntu 16.04 (xenial)	7.0
Ubuntu 14.04 (trusty)	5.5
Raspbian 10 (buster)	7.3
Raspbian 8 (stretch)	7.0

Nas distribuições stretch, jessie, xenial e trusty, a dependência do PHP 7.2 não está disponível, e portanto o Zabbix Frontend 5.0 ou versões mais recentes não podem ser facilmente instaladas. Considerando isto, o pacote `zabbix-frontend-php` foi substituído pelo pacote `zabbix-frontend-php-deprecated` nas distribuições mencionadas.

A principal diferença é a ausência de dependências diretas de quaisquer pacotes do PHP ou Web Servers. Deste modo, o usuário pode (e deve) fornecer estas dependências por conta própria. Em outras palavras, instalando o pacote `zabbix-frontend-php-deprecated` por conta, não lhe dará um Frontend funcional. Um Web Server, assim como o PHP 7.2 com seus módulos devem ser instalados manualmente (usando PPAs / compilando o PHP a partir das fontes). Nós não aprovamos nenhum método particular de instalação.

Note:

A maneira oficial de obter o PHP 7.2 ou mais recente em versões antigas do Debian/Ubuntu é atualizando o sistema para buster/bionic.

Os módulos PHP necessários para o Zabbix Frontend são `php-gd`, `php-bcmath`, `php-mbstring`, `php-xml`, `php-ldap` e `php-json`.

7 Atualizar procedimento

Visão geral

Esta seção provê informações de atualização para o Zabbix **6.0**:

- usando pacotes:
 - para [Red Hat Enterprise Linux/CentOS](#)
 - para [Debian/Ubuntu](#)
- usando [fontes](#)

Atualização direta para o Zabbix 6.0.x é possível a partir de Zabbix **5.4.x**, **5.2.x**, **5.0.x**, **4.4.x**, **4.2.x**, **4.0.x**, **3.4.x**, **3.2.x**, **3.0.x**, **2.4.x**, **2.2.x** e **2.0.x**. Para atualização a partir de versões mais antigas consulte a documentação para Zabbix 2.0 e versões anteriores.

Note:

Esteja ciente de que, após a atualização, algumas integrações software de terceiros no Zabbix podem ser afetadas, caso o software externo não seja compatível com a versão atualizada de Zabbix.

Atualização a partir de pacotes

Visão geral

Esta seção provê os passos necessários para uma [atualização](#) bem-sucedida utilizando os pacotes oficiais RPM e DEB fornecidos pela Zabbix para:

- [Red Hat Enterprise Linux/CentOS](#)
- [Debian/Ubuntu](#)

Pacotes do Zabbix nos repositórios de SO

Frequentemente, as distribuições de SO (em particular, distribuições baseadas em Debian) proveem seus próprios pacotes do Zabbix.

Note que estes pacotes não são suportados pela Zabbix, eles são tipicamente desatualizados e carecem de funcionalidades recentes e correções de problemas. Apenas os pacotes do repo.zabbix.com são oficialmente suportados.

Se você estiver atualizando a partir de pacotes fornecidos pelas distribuições de SO (ou os tenha instalado em algum momento), siga este procedimento para alterar seu ambiente para os pacotes oficiais da Zabbix:

1. Sempre desinstale os pacotes antigos em primeiro lugar.
2. Faça uma verificação por arquivos que possam ter sido deixados para trás depois da desinstalação.
3. Instale os pacotes oficiais fornecidos pela Zabbix seguindo [as instruções de instalação](#).

Nunca faça uma atualização direta, pois isso pode resultar em uma instalação inutilizada.

1 Red Hat Enterprise Linux

Visão geral

Esta seção oferece as etapas necessárias para uma atualização de sucesso **upgrade** do Zabbix **5.4.x** para o Zabbix **6.0.x**, usando os pacotes oficiais do Zabbix para Red Hat Enterprise Linux.

Embora a atualização dos agentes Zabbix não seja obrigatória (mas recomendada), o servidor e proxies Zabbix devem estar na mesma versão **same major version**. Portanto, em uma configuração servidor-proxy, o servidor e todos os proxies Zabbix devem ser pausados e atualizados. Manter os proxies em execução durante a atualização do servidor não trará nenhum benefício, pois durante a atualização do proxy, os dados antigos serão descartados e nenhum dado novo será coletado até que a configuração do proxy seja sincronizada com o servidor.

Observe que, com um banco de dados SQLite em proxies, os dados de histórico dos proxies anteriores à atualização serão perdidos, pois a atualização do banco de dados SQLite não é suportada e o arquivo de banco de dados SQLite deve ser removido manualmente. Quando o proxy é iniciado pela primeira vez e o arquivo de banco de dados SQLite está ausente, o proxy o cria automaticamente.

Dependendo do tamanho do banco de dados, sua atualização para a versão 6.0 pode levar um bom tempo.

Warning:

Antes de iniciar a atualização, leia as notas de atualização relevantes.

Notas de atualização!

As seguintes notas de atualização estão disponíveis:

Upgrade from	Leia todas as notas de atualização	As mudanças mais importantes entre as versões
5.4.x	Para: Zabbix 6.0	As versões mínimas do banco de dados necessários foram aumentadas; O Servidor/proxy não irá iniciar se o banco de dados estiver desatualizado; Registros de log de auditoria perdidos devido a mudança na estrutura do banco de dados.
5.2.x	Para: Zabbix 5.4 Zabbix 6.0	As versões mínimas do banco de dados necessários foram aumentadas; Itens agregados removidos como um tipo separado.
5.0.x LTS	Para: Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Versão mínima do PHP exigida aumentada de 7.2.0 para 7.2.5.
4.4.x	Para: Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	O suporte do IBM DB2 foi descontinuado; Versão mínima do PHP exigida aumentada de 5.4.0 para 7.2.0; As versões mínimas requeridas de banco de dados foram aumentadas; O diretório de arquivo PHP do Zabbix foi alterado.

Upgrade from	Leia todas as notas de atualização	As mudanças mais importantes entre as versões
4.2.x	Para: Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Os tipos de mídia Jabber e Ez Texting foram removidos.
4.0.x LTS	Para: Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Proxies antigos não conseguem mais reportar dados para um servidor atualizado; Agentes mais novos não serão mais capazes de funcionar com um servidor Zabbix antigo.
3.4.x	Para: Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Bibliotecas 'libpthread' e 'zlib' agora são obrigatórias; O suporte para o protocolo de texto simples foi removido e o cabeçalho é obrigatório; Os agentes Zabbix versão Pre-1.4 não são mais suportados; Agora, o parâmetro do servidor na configuração do proxy passivo é obrigatório.
3.2.x	Para: Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	O suporte ao SQLite e o banco de dados de backend foram removidos para o servidor/frontend do Zabbix; Expressões regulares compatíveis com Perl (PCRE) são suportadas em vez de POSIX estendido; as bibliotecas 'libpcre' e 'libevent' são obrigatórias para o servidor Zabbix; Verificações de código de saída para os parâmetros de usuário foram adicionadas, comandos remotos e itens system.run[] sem a flag 'nowait' assim como os scripts executados pelo servidor Zabbix; Zabbix Java gateway precisa ser atualizado para suportar nova funcionalidade.
3.0.x LTS	Para: Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	A atualização do banco de dados pode ser lenta, dependendo do tamanho da tabela do histórico.
2.4.x	Para: Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	A versão mínima do PHP requerida foi aumentada de 5.3.0 para 5.4.0 O parâmetro LogFile do agente deve ser especificado.

Upgrade from	Leia todas as notas de atualização	As mudanças mais importantes entre as versões
2.2.x LTS	Para: Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Monitoramento distribuído baseado em nós removido.
2.0.x	Para: Zabbix 2.2 Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Versão mínima do PHP aumentada de 5.1.6 para 5.3.0; Um banco de dados MySQL que diferencia maiúsculas de minúsculas é necessário para o correto funcionamento do servidor; o conjunto de caracteres utf8 e a ordenação utf8_bin são necessários para que o servidor Zabbix funcione corretamente com o banco de dados MySQL. Ver database creation scripts . Extensão 'mysqli' PHP é necessária em vez de 'mysql'.

Você também pode desejar verificar o [requirements](#) para 6.0.

Note:

Pode ser útil executar duas sessões SSH paralelas durante a atualização, realizando as etapas de atualização em uma e monitorando os logs de servidor/proxy em outra. Por exemplo, execute `tail -f zabbix_server.log` ou `tail -f zabbix_proxy.log` na segunda sessão do SSH, mostrando os registros mais recentes do arquivos de log e possíveis erros em tempo real. Isso pode ser crítico para instâncias em produção.

Procedimento de atualização

1 Pausar os processos Zabbix

Pausar o servidor Zabbix para garantir que nenhum dado novo seja inserido no banco de dados.

```
systemctl stop zabbix-server
```

If upgrading the proxy, stop proxy too.

```
systemctl stop zabbix-proxy
```

Attention:

It is no longer possible to start the upgraded server and have older and unupgraded proxies report data to a newer server. This approach, which was never recommended nor supported by Zabbix, now is officially disabled, as the server will ignore data from unupgraded proxies.

2 Back up do banco de dados existente Zabbix

Esta é uma etapa muito importante. Assegure-se de que você tenha um backup do seu banco de dados. Isso ajudará caso o procedimento de atualização falhe (falta de espaço em disco, desligamento, qualquer problema inesperado).

3 Backup dos arquivos de configuração, arquivos PHP e binários Zabbix

Realize uma cópia do backup binários Zabbix, dos arquivos de configuração e do diretório de arquivos PHP.

Arquivos de configuração:

```
mkdir /opt/zabbix-backup/
cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
```

```
cp /etc/httpd/conf.d/zabbix.conf /opt/zabbix-backup/
```

Arquivo de PHP files e binários Zabbix:

```
cp -R /usr/share/zabbix/ /opt/zabbix-backup/  
cp -R /usr/share/zabbix-* /opt/zabbix-backup/
```

4 Atualização do pacote de configuração do repositório

Antes de prosseguir com a atualização, atualizar seu pacote do repositório atual. Em RHEL 9, execute:

```
rpm -Uvh https://repo.zabbix.com/zabbix/6.0/rhel/9/x86_64/zabbix-release-latest.el9.noarch.rpm
```

(Para versões RHEL anteriores, substituir o link acima pelo link correto a partir do [Zabbix repository](#)).

5 Atualizar componentes Zabbix

Para atualizar os componentes Zabbix, você pode executar algo semelhante a:

```
dnf upgrade zabbix-server-mysql zabbix-web-mysql zabbix-agent
```

Caso esteja utilizando PostgreSQL, substitua `mysql` por `pgsql` no comando. Caso esteja atualizando o proxy, substitua o `server` por `proxy` no comando. Caso esteja atualizando o agente 2, substitua o `zabbix-agent` por `zabbix-agent2` no comando.

Atualizar o agente 2 Zabbix com o comando `dnf install zabbix-agent2`, pode levar a um erro. Para mais informações, ver [Known issues](#).

Para atualizar o web frontend com Apache **on RHEL 8** corretamente, execute também:

```
dnf install zabbix-apache-conf
```

6 Revisar parâmetros de configuração do componente

Assegure-se de revisar as notas de atualização [Upgrade notes](#) para verificar se alguma mudança dos parâmetros de configuração é necessária.

7 Inicie os processos Zabbix

Inicie a atualização dos componentes Zabbix.

```
systemctl start zabbix-server  
systemctl start zabbix-proxy  
systemctl start zabbix-agent  
systemctl start zabbix-agent2
```

Limpar os cookies e cache do navegador

Após a atualização, pode ser necessário limpar os cookies e cache do navegador para que a interface do Zabbix funcione corretamente.

Atualização entre versões menores

É possível fazer atualização entre versões menores de 6.0.x (por exemplo, de 6.0.1 para 6.0.3). Fazer atualização entre as versões menores é simples.

Para executar a atualização da versão menor do Zabbix, é necessário executar:

```
sudo dnf upgrade 'zabbix-*'
```

Para executar a atualização da versão menor do servidor Zabbix, execute:

```
sudo dnf upgrade 'zabbix-server-*'
```

Para executar a atualização do agente da versão menor do Zabbix, execute:

```
sudo dnf upgrade 'zabbix-agent-*'
```

ou, para o agente 2 do Zabbix:

```
sudo dnf upgrade 'zabbix-agent2-*'
```

2 Debian/Ubuntu

Visão geral

Esta seção fornece os passos necessários para uma atualização bem-sucedida do [upgrade](#) do Zabbix **5.4.x** para o Zabbix **6.0.x** usando os pacotes oficiais do Zabbix para Debian/Ubuntu. Embora a atualização dos agentes do Zabbix não seja obrigatória (mas recomendada), o servidor e os proxies do Zabbix devem ser da mesma versão principal [same major version](#). Portanto,

em uma configuração de servidor-proxy, o servidor Zabbix e todos os proxies devem ser interrompidos e atualizados. Manter os proxies em execução durante a atualização do servidor não trará mais nenhum benefício, pois durante a atualização do proxy seus dados antigos serão descartados e nenhum novo dado será coletado até que a configuração do proxy seja sincronizada com o servidor. Observe que com o banco de dados SQLite nos proxies, os dados de histórico dos proxies antes da atualização serão perdidos, porque a atualização do banco de dados SQLite não é suportada e o arquivo de banco de dados SQLite deve ser removido manualmente. Quando o proxy é iniciado pela primeira vez e o arquivo de banco de dados SQLite está ausente, o proxy o cria automaticamente.

Dependendo do tamanho do banco de dados, a atualização do banco de dados para a versão 6.0 pode levar bastante tempo.

Warning:

Antes da atualização, certifique-se de ler as **notas de atualização!**

As seguintes notas de atualização estão disponíveis:

Atualização a partir de	Leia as notas de atualização completas	Mudanças mais importantes entre versões
5.4.x	Para: Zabbix 6.0	Versões mínimas necessárias do banco de dados aumentadas; O Server/proxy não será iniciado se o banco de dados estiver desatualizado; Registros de log de auditoria perdidos devido à mudança na estrutura do banco de dados.
5.2.x	Para: Zabbix 5.4 Zabbix 6.0	Versões mínimas necessárias do banco de dados aumentadas; tens de agregação removidos como um tipo separado.
5.0.x LTS	Para: Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Versão mínima necessária do PHP aumentada de 7.2.0 para 7.2.5.
4.4.x	Para: Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Suporte ao IBM DB2 removido; versão mínima necessária do PHP aumentada de 5.4.0 para 7.2.0; versões mínimas necessárias do banco de dados aumentadas; Diretório de arquivos PHP do Zabbix alterado..
4.2.x	Para: Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Tipos de mídia Jabber e Ez Texting removidos..
4.0.x LTS	Para: Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Proxies mais antigos não poderão mais enviar dados para um servidor atualizado; Agentes mais recentes não serão mais capazes de funcionar com um servidor Zabbix mais antigo.
3.4.x	For: Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Bibliotecas 'libpthread' e 'zlib' agora são obrigatórias; Suporte para protocolo de texto simples removido e o cabeçalho é obrigatório; Versões pré-1.4 do Zabbix agentes não são mais suportadas; O parâmetro do servidor na configuração do proxy passivo agora é obrigatório.

Atualização a partir de	Leia as notas de atualização completas	Mudanças mais importantes entre versões
3.2.x	Para: Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Suporte do SQLite como banco de dados backend removido para servidor/frontend do Zabbix; Expressões regulares compatíveis com Perl (PCRE) suportadas em vez de POSIX estendido; 'Bibliotecas 'libpcre' e 'libevent' obrigatórias para o servidor Zabbix;
Verificações de código de saída adicionadas para parâmetros de usuário, comandos remotos e itens system.run [] do Zabbix server sem a flag 'nowait', bem como scripts executados pelo servidor Zabbix; O Java gateway do Zabbix deve ser atualizado para suportar novas funcionalidades.		
3.0.x LTS	Para: Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	A atualização do banco de dados pode ser lenta, dependendo do tamanho da tabela de histórico.
2.4.x	Para: Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Aumento da versão mínima do PHP de 5.3.0 para 5.4.0 O parâmetro do agente LogFile deve ser especificado
2.2.x LTS	For: Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Monitoramento distribuído baseado em nós removido

Atualização a partir de	Leia as notas de atualização completas	Mudanças mais importantes entre versões
2.0.x	Para: Zabbix 2.2 Zabbix 2.4 Zabbix 3.0 Zabbix 3.2 Zabbix 3.4 Zabbix 4.0 Zabbix 4.2 Zabbix 4.4 Zabbix 5.0 Zabbix 5.2 Zabbix 5.4 Zabbix 6.0	Aumento da versão mínima do PHP de 5.1.6 para 5.3.0;; Banco de dados MySQL com distinção entre maiúsculas e minúsculas necessário para o funcionamento adequado do servidor; o conjunto de caracteres utf8 e a colação utf8_bin são necessários para o servidor Zabbix funcionar corretamente com o banco de dados MySQL. Consulte os scripts database creation scripts . Extensão PHP 'mysqli' necessária em vez de 'mysql'

Você também pode querer verificar os requisitos [requirements](#) para o 6.0.

Note:

Pode ser útil executar duas sessões SSH paralelas durante a atualização, executando os passos de atualização em uma e monitorando os logs do servidor/proxy em outra. Por exemplo, execute `tail -f zabbix_server.log` ou `tail -f zabbix_proxy.log` na segunda sessão SSH, mostrando as últimas entradas de log e possíveis erros em tempo real. Isso pode ser crítico para instâncias de produção.

Procedimento de atualização

1 Pare os processos do Zabbix

Pare o servidor Zabbix para garantir que nenhum dado novo seja inserido no banco de dados.

```
service zabbix-server stop
```

Se estiver atualizando o proxy do Zabbix, pare o proxy também.

```
service zabbix-proxy stop
```

2 Fazer backup do banco de dados do Zabbix

Este é um passo muito importante. Certifique-se de ter um backup do seu banco de dados. Isso ajudará caso o procedimento de atualização falhe (falta de espaço em disco, desligamento inesperado, qualquer problema inesperado).

3 Backup dos arquivos de configuração, arquivos PHP e binários do Zabbix

Faça uma cópia de backup dos binários do Zabbix, arquivos de configuração e do diretório de arquivos PHP.

Arquivos de configuração:

```
# mkdir /opt/zabbix-backup/
# cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
# cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-backup/
```

Arquivos PHP e binários do Zabbix:

```
# cp -R /usr/share/zabbix/ /opt/zabbix-backup/
# cp -R /usr/share/doc/zabbix-* /opt/zabbix-backup/
```

4 Atualizar pacote de configuração do repositório

Para prosseguir com a atualização, seu pacote de repositório atual precisa ser desinstalado.

```
rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Em seguida, instale o novo pacote de configuração do repositório.

No **Debian 12**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release_6.0-5+debian12_all.deb
dpkg -i zabbix-release_6.0-5+debian12_all.deb
```

No **Debian 11**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release_6.0-1+debian11_all.deb
dpkg -i zabbix-release_6.0-1+debian11_all.deb
```

No **Debian 10**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release_6.0-1+debian10_all.deb
dpkg -i zabbix-release_6.0-1+debian10_all.deb
```

On **Debian 9** run:

```
wget https://repo.zabbix.com/zabbix/6.0/debian/pool/main/z/zabbix-release/zabbix-release_6.0-1+debian9_all.deb
dpkg -i zabbix-release_6.0-1+debian9_all.deb
```

No **Ubuntu 20.04**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.0-1+ubuntu20.04_all.deb
dpkg -i zabbix-release_6.0-1+ubuntu20.04_all.deb
```

On **Ubuntu 18.04** run:

```
wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.0-1+ubuntu18.04_all.deb
dpkg -i zabbix-release_6.0-1+ubuntu18.04_all.deb
```

No **Ubuntu 16.04**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.0-1+ubuntu16.04_all.deb
dpkg -i zabbix-release_6.0-1+ubuntu16.04_all.deb
```

No **Ubuntu 14.04**, execute:

```
wget https://repo.zabbix.com/zabbix/6.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_6.0-1+ubuntu14.04_all.deb
dpkg -i zabbix-release_6.0-1+ubuntu14.04_all.deb
```

Atualize as informações do repositório.

```
apt-get update
```

5 Atualizar os componentes do Zabbix

Para atualizar os componentes do Zabbix, você pode executar algo como:

```
apt-get install --only-upgrade zabbix-server-mysql zabbix-frontend-php zabbix-agent
```

Se estiver usando PostgreSQL, substitua `mysql` por `pgsql` no comando. Se estiver atualizando o proxy, substitua `server` por `proxy` no comando. Se estiver atualizando o Zabbix agent 2, substitua `zabbix-agent` por `zabbix-agent2` no comando.

Attention:

Atualizar o Zabbix agent 2 com o comando `apt install zabbix-agent2` pode resultar em um erro. Para mais informações, consulte [Known issues](#).

Em seguida, para atualizar corretamente o frontend da web com o Apache, execute também:

```
apt-get install zabbix-apache-conf
```

Distribuições **anteriores ao Debian 10 (buster) / Ubuntu 18.04 (bionic) / Raspbian 10 (buster)** não fornecem PHP 7.2 ou mais recente, que é necessário para o frontend do Zabbix 6.0. Consulte informações [information](#) sobre como instalar o frontend do Zabbix em distribuições mais antigas.

6 Revisar parâmetros de configuração dos componentes

Certifique-se de revisar as [Upgrade notes](#) para verificar se há alguma alteração nos parâmetros de configuração necessários.

7 Inicie os processos do Zabbix

Inicie os componentes do Zabbix atualizados.

```
# service zabbix-server start
# service zabbix-proxy start
# service zabbix-agent start
# service zabbix-agent2 start
```

8 Limpe os cookies e cache do navegador

Após a atualização pode ser necessário limpar os cookies e o cache do navegador para que a interface web do Zabbix funcione corretamente.

Atualização entre versão secundárias

É possível atualizar versões secundárias do Zabbix 6.0.x (por exemplo, de 6.0.1 para 6.0.3). A atualização entre versões secundárias é fácil.

Para atualizar a versão secundária do Zabbix, execute:

```
$ sudo apt install --only-upgrade 'zabbix.*'
```

Para atualizar a versão secundária do Zabbix Server, execute:

```
$ sudo apt install --only-upgrade 'zabbix-server.*'
```

Para atualizar a versão secundária do Zabbix Agent, utilize:

```
$ sudo apt install --only-upgrade 'zabbix-agent.*'
```

ou, para Zabbix Agent 2:

```
$ sudo apt install --only-upgrade 'zabbix-agent2.*'
```

Atualização a partir dos fontes

Visão geral

Esta seção fornece as etapas necessárias para uma **atualização** do Zabbix **5.4.x** para o Zabbix **6.0.x** usando fontes oficiais do Zabbix.

Embora atualizar os agentes Zabbix não seja obrigatório (mas recomendado), o Zabbix servidor e proxies devem ser da **mesma versão**. Portanto, em uma configuração de servidor-proxy, servidor Zabbix e todos os proxies devem ser interrompidos e atualizados. Manter os proxies em execução não trará mais nenhum benefício bem como durante a atualização os dados antigos serão descartados e nenhum dado novo será reunidos até que a configuração do proxy seja sincronizada com o servidor.

Attention:

Não é mais possível iniciar o upgrade do servidor e ter proxies mais antigos, os mesmos não relatam dados para o novo servidor. Esta abordagem, que nunca foi recomendada nem suportada pelo Zabbix, agora está oficialmente desabilitada, pois o servidor irá ignorar dados de proxies não atualizados.

Observe que com o banco de dados SQLite em proxies, os dados históricos de proxies antes da atualização serão perdidos, porque a atualização do banco de dados SQLite não é suportada e o arquivo de banco de dados SQLite deve ser removido manualmente. Quando o proxy é iniciado pela primeira vez e o arquivo de banco de dados SQLite é ausente, o proxy o cria automaticamente.

Dependendo do tamanho do banco de dados, a atualização do banco de dados para a versão 6.0 pode demorar muito tempo.

Warning:

Antes da atualização, certifique-se de ler as **notas de atualização!**

As seguintes notas de atualização estão disponíveis:

[Atualizar de] Leia as notas completas de atualização | Mudanças mais importantes entre as versões | |-----|-----|-----|
-----| 5.4.x | Para:
Zabbix 6.0 | Versões mínimas necessárias do banco de dados atualizadas;
O servidor/proxy não será iniciado se o banco de dados estiver desatualizado;
Registros de log de auditoria perdidos porque de alteração da estrutura do banco de dados. | 5.2.x | Para:
Zabbix 5.4
Zabbix 6.0 | Versões de banco de dados mínimas exigidas aumentadas;
Itens agregados removidos como um tipo separado. | 5.0.x LTS | Para:
Zabbix 5.2
Zabbix 5.4
Zabbix 6.0 | Versão mínima exigida do PHP aumentada de 7.2.0 para 7.2.5. | 4.4.x | Para:
Zabbix 5.0
Zabbix 5.2
Zabbix 5.4
Zabbix 6.0 | Suporte do IBM DB2 caiu;
Versão mínima exigida do PHP aumentada de 5.4.0 para 7.2.0;
Versões mínimas exigidas do banco de dados aumentadas;
Diretório de arquivo PHP Zabbix alterado. | 4.2.x | Para:
Zabbix 4.4
Zabbix 5.0
Zabbix 5.2
Zabbix 5.4
Zabbix 6.0 | Jabber, Ez Tipos de mídia de texto removidos. | 4.0.x LTS | Para:
Zabbix 4.2
Zabbix 4.4
Zabbix 5.0
Zabbix 5.2
Zabbix 5.4
Zabbix 6.0 | Proxies mais antigos não podem mais reportar dados para um servidor atualizado;
Agentes mais novos não poderão mais trabalhar com um servidor Zabbix mais antigo. | 3.4.x | Para:
Zabbix 4.0
Zabbix 4.2
Zabbix 4.4
Zabbix 5.0
Zabbix 5.2
Zabbix 5.4
Zabbix 6.0 | Bibliotecas 'libpthread' e 'zlib' agora obrigatório;
Suporte para protocolo de texto simples descartado e cabeçalho é obrigatório;
Os agentes Zabbix da versão pré-1.4 não são mais suportados;

Processo de atualização do Server

1 Stop server

Pare o Zabbix Server para certificar-se de que nenhum novo dado está sendo gravado no banco de dados.

2 Faça backup do banco de dados Zabbix

Este é um passo muito importante. Certifique-se de que você tem um backup do seu banco de dados. Ele o ajudará se o procedimento de atualização falhar (falta de espaço em disco, desligamento forçado, ou qualquer outro problema inesperado).

3 Backup dos arquivos de configuração, arquivos PHP e binários do Zabbix

Faça uma cópia de backup dos binários do Zabbix, arquivos de configuração e do diretório de arquivos PHP.

4 Instale os novos binários do Server

Utilize estas [instruções](#) para compilar o Zabbix Server a partir dos fontes.

5 Revise os parâmetros de configuração do Server

Consulte as notas de atualização para detalhes sobre [alterações obrigatórias](#).

Para novos parâmetros opcionais, veja a seção [O que há de novo](#).

6 Inicie os novos binários do Zabbix

Inicie os novos binários. Verifique os arquivos de log para garantia de que os binários subiram corretamente.

O Zabbix Server atualizará automaticamente o banco de dados. Quando iniciando, o Zabbix Server informa as versões de banco de dados atual (mandatório e opcional) e exigida. Se a versão mandatória atual é mais antiga que a versão exigida, o Zabbix Server executa as atualizações do banco de dados necessárias de forma automática. O andamento do processo de atualização (porcentagem) é registrado no arquivo de log do server. Quando a atualização estiver completa, uma mensagem de "atualização de banco de dados completa" é registrada no log. Se algum dos passos de atualização falhar, o Zabbix Server não inicializará. O Zabbix Server também não iniciará se a versão de banco de dados atual for maior que a versão requerida. O Zabbix Server iniciará apenas se a versão atual corresponder à versão necessária.

```
8673:20161117:104750.259 current database version (mandatory/optional): 03040000/03040000
8673:20161117:104750.259 required mandatory version: 03040000
```

Antes de você iniciar o Server:

- Certifique-se de que o usuário do banco de dados possui permissões suficientes (create table, drop table, create index, drop index)
- Certifique-se de que tenha espaço em disco suficiente
- .

7 Instale a nova interface web do Zabbix

A versão mínima do PHP exigida é a 7.2.5. Atualize se necessário e siga as [instruções de instalação](#).

8 Limpe os cookies e cache do navegador

Após a atualização pode ser necessário limpar os cookies e o cache do navegador para que a interface web do Zabbix funcione corretamente.

Processo de atualização do Proxy

1 Pare o Proxy

Para continuar pare o Zabbix Proxy.

2 Backup da configuração e binários do Zabbix Proxy

Faça uma cópia de backup do arquivo de configuração e dos binários do Zabbix Proxy.

3 Instale novo binários do Proxy

Use estas [instruções](#) para compilar Zabbix Proxy a partir dos fontes.

4 Revise os parâmetros de configuração do Proxy

Não há alterações obrigatórias aos [parâmetros](#) do Zabbix Proxy nesta versão.

5 Inicie o novo ZabbixProxy

Inicie o novo Zabbix Proxy. Consulte os arquivos de log para garantir que Proxy iniciou corretamente.

O Zabbix proxy atualizará automaticamente o banco de dados. A atualização do banco de dados ocorrerá de maneira semelhante à inicialização do [Zabbix Server](#).

Processo de atualização do Agent

Attention:

A atualização dos Agents não é obrigatória. Você só precisa atualizar os Agents se exigido para acessar novas funcionalidades.

O processo de atualização descrito nesta seção pode ser usado para atualização tanto do Zabbix Agent quanto do Zabbix Agent 2.

1 Pare o Agent

Para prosseguir com a atualização pare o Zabbix Agent.

2 Backup da configuração e binários do Zabbix Agent

Faça uma cópia de backup do arquivo de configuração e dos binários do Zabbix Agent.

3 Instale novos binários do Agent

Use estas [instruções](#) para compilar o Zabbix Agent a partir dos fontes.

Alternativamente, você pode baixar os Zabbix Agents pré-compilados da [página de download da Zabbix](#).

4 Revise os parâmetros de configuração do Agent

Não há alterações obrigatórias nos parâmetros desta versão do **Agent** e nem do **Agent 2**.

5 Inicie o novo Zabbix Agent

Inicie o novo Zabbix Agent. Confirme nos arquivos de log se o Agent iniciou corretamente.

Atualização entre versão secundárias (minor versions)

Quando atualizando entre versões secundárias do Zabbix 6.0.x (por exemplo de 6.0.1 para 6.0.3) é necessário executar as mesmas ações para o Zabbix Server/Proxy/Agent assim como durante a atualização entre versões primárias (major versions). A única diferença é que quando atualizando entre versões secundárias nenhuma alteração de banco de dados é feita.

7 Modificações nos Templates

Esta página lista todas as modificações nos templates padrões que são fornecidos com o Zabbix.

Perceba que realizando o upgrade da versão mais atual do Zabbix, não será realizado o upgrade automaticamente dos templates usados. É recomendado modificar os templates em uma instalação existente por:

- Baixar os templates mais recentes no [Zabbix Gitrepository](#); -Então, em Configurações → Templates você pode importar eles manualmente para dentro do Zabbix. Se já existirem templates com o mesmo nome, deve ser marcada a opção delete missing durante a importação para garantir uma importação correta. Dessa forma, itens antigos que não pertencem ao novo template atualizado serão excluídos (perceba que isso implicará na perda do histórico desses itens antigos).

Mudanças no Zabbix 6.0.0**Templates novos**

Consulte a list de templates novo [new templates](#) na versão Zabbix 6.0.0.

Mudanças nos templates

- A macro {#FSLABEL} foi adicionada aos itens nomes e descrições correspondentes nos templates Windows by Zabbix agent and Windows by Zabbix agent active
- O item agente vfs.file.cksum[/etc/passwd] foi modificado para vfs.file.cksum[/etc/passwd,sha256]
- Uma nova verificação do Zabbix [process,odbc poller,avg,busy] foi adicionada aos templates Zabbix server, Zabbix proxy, Remote Zabbix server e Remote Zabbix proxy. A métrica é usada para monitorar o tempo médio pelo qual os processos ODBC estiveram ocupados durante o último minuto (em porcentagem).

MUDANÇAS NO Zabbix 6.0.2

Agora o template Generic Java JMX contém duas regras de descoberta:

- Descoberta do coletor de lixo
- Descoberta da memória pool

MUDANÇAS NO Zabbix 6.0.3

Um template novo OpenWeatherMap by HTTP está disponível.

As seguintes alterações foram feitas nos templates existentes:

- Nos templates Windows services by Zabbix agent, Windows services by Zabbix agent active, Windows by Zabbix agent, Windows by Zabbix agent active {\$SERVICE.NAME.NOT_MATCHES} o valor da macro foi atualizado para remover uma lista extensa de serviços.
- Agora o template PostgreSQL by Zabbix agent 2 verificará o número de consultas lentas e irá gerar um problema caso a média exceder um limite.

MUDANÇAS NO Zabbix 6.0.4

Novos templates estão disponíveis:

- TrueNAS SNMP - monitoramento do sistema de armazenamento pelo SNMP
- Proxmox VE by HTTP - consulte as instruções de configuração para [HTTP templates](#)

Os templates SMART by Zabbix agent 2 e SMART by Zabbix agent 2 (active) foram atualizados:

- A regra LLD Attribute discovery LLD foi excluída, enquanto a regra LLD Disk discovery agora descobrirá discos com base no conjunto predefinido de atributos específicos do fabricante;
- Agora o item **smart.disk.get** consegue retornar apenas informação sobre um disco específico, em vez de todos os discos.

Novas macros foram adicionadas aos modelos, permitindo definir limites de advertência e críticos para a utilização do sistema de arquivo no monitoramento de sistema de arquivos virtuais

HOST-RESOURCES-MIB storage SNMP, Linux by Prom, Linux filesystems SNMP, Linux filesystems by Zabbix agent active, Linux filesystems by Zabbix agent, Mellanox SNMP, PFSense SNMP, Windows filesystems by Zabbix agent active, Windows filesystems by Zabbix agent.

A utilização de triggers para sistemas de arquivo foram atualizados para utilizar essa macros.

MUDANÇAS NO Zabbix 6.0.5

Novos templates estão disponíveis:

- CockroachDB por HTTP
- Envoy Proxy por HTTP
- HashiCorp Consul Cluster por HTTP
- HashiCorp Consul Node por HTTP

Consulte as instruções de configuração para templates [HTTP templates](#).

MUDANÇAS NO Zabbix 6.0.6

Novos templates estão disponíveis:

- HPE MSA 2040 Storage por HTTP
- HPE MSA 2060 Storage por HTTP
- HPE Primera por HTTP

Consulte as instruções de configuração para [HTTP templates](#).

O template [PostgreSQL Agent 2 template](#) foi atualizado:

Foi adicionado um trigger para detectar falhas de soma de verificação ao item DBstat do template PostgreSQL Agent 2. De acordo com a documentação [PostgreSQL documentation](#), você pode usar a soma de verificação em páginas de dados para ajudar a detectar corrupção causada pelo sistema I/O que, de outra forma, seria silenciosa.

MUDANÇAS NO Zabbix 6.0.7

Um novo template [template](#) HPE Synergy por HTTP está disponível.

Agora os templates HashiCorp Consul Node por HTTP e HashiCorp Consul Cluster por HTTP suportam Consul namespaces.

MUDANÇAS NO Zabbix 6.0.8

Um novo template [template](#) OPNsense by SNMP está disponível.

MUDANÇAS NO Zabbix 6.0.13

Novos templates estão disponíveis:

- AWS EC2 por HTTP
- AWS por HTTP
- AWS RDS instance por HTTP
- AWS S3 bucket por HTTP
- Azure por HTTP
- Control-M server por HTTP

- Control-M enterprise manager por HTTP
- Veeam Backup Enterprise Manager por HTTP
- Veeam Backup and Replication por HTTP

Consulte as instruções de configuração para [HTTP templates](#).

O template [Oracle by Zabbix agent 2](#) foi atualizado:

- Os seguintes itens estáticos, que solicitavam dados para todos os objetos relevantes BD em uma única consulta, foram removidos:
 - "Oracle: Get archive log info"
 - "Oracle: Get ASM stats"
 - "Oracle: Get CDB and No-CDB info"
 - "Oracle: Get PDB info"
 - "Oracle: Get tablespaces stats"
- O seguintes protótipos de itens do agente foram adicionados às correspondentes regras de descoberta:
 - Regra de descoberta do Archive log: "Archivelog '{#DEST_NAME}': Get archive log info"
 - Descoberta do ASM disk groups: "ASM '{#DGNAME}': Get ASM stats"
 - Descoberta do banco de dados: "Oracle Database '{#DBNAME}': Get CDB and No-CDB info"
 - Descoberta de PDB: "Oracle Database '{#DBNAME}': Get PDB info"
 - Descoberta do Tablespace "Oracle TBS '{#TABLESPACE}': Get tablespace stats"

MUDANÇAS NO Zabbix 6.0.18

Um novo template [template Google Cloud Platform by HTTP \(GCP by HTTP\)](#) está disponível.

Consulte as instruções de configuração para [HTTP templates](#).

Agora o template Azure by HTTP também funciona com Azure Cosmos DB para MongoDB.

MUDANÇAS NO Zabbix 6.0.20

Novos templates estão disponíveis:

- [AWS ECS Cluster by HTTP](#) (junto com seu [Serverless Cluster version](#))
- [Cisco SD-WAN by HTTP](#)
- [OpenStack by HTTP](#), que incluem o template OpenStack Nova by HTTP para monitoramento do serviço OpenStack Nova
- [PostgreSQL by ODBC](#)

MUDANÇAS NO Zabbix 6.0.21

Um novo template está disponível:

- [AWS Cost Explorer by HTTP](#)

MUDANÇAS NO Zabbix 6.0.22

Novos templates estão disponíveis:

- [Acronis Cyber Protect Cloud by HTTP](#)
- [HashiCorp Nomad by HTTP](#)
- [MantisBT by HTTP](#)

MUDANÇAS NO Zabbix 6.0.23

Novos templates

Novos templates estão disponíveis:

- [FortiGate by HTTP](#)
- [FortiGate by SNMP](#)
- [Nextcloud by HTTP](#)

Templates atualizados

- Agora os templates [PostgreSQL by ODBC](#) e [PostgreSQL by Zabbix agent 2](#) incluem o item e o trigger para monitoramento PostgreSQL version.
- O template [Cisco Meraki organization by HTTP](#) foi complementado com itens, protótipos de itens, regras de LLD e macros relacionadas à autenticação, licenças, networks, papéis SAML, e status de VPN.

MUDANÇAS NO Zabbix 6.0.24

Novos templates

Um template novo está disponível:

- [HPE iLO by HTTP](#)

Templates atualizados

Integração com OpenShift foi adicionada ao template [Kubernetes cluster state by HTTP](#).

MUDANÇAS NO Zabbix 6.0.26

Novos templates

Um conjunto de templates [Azure by HTTP](#) foi complementado com o template Azure Cost Management by HTTP template.

Templates atualizados

O template [MSSQL by ODBC](#) :

- Um item novo foi adicionado - MSSQL DB '#{DBNAME}': Recovery model, que retorna o modelo de recuperação do banco de dados durante a descoberta do banco de dados;
- novas macros, nomeadas; {\$MSSQL.BACKUP_FULL.USED}, {\$MSSQL.BACKUP_DIFF.USED}, {\$MSSQL.BACKUP_LOG.USED}, foram adicionadas - elas podem ser usadas para desabilitar os triggers de idade do backup para um determinado banco de dados;
- Regras de descoberta LLD para quórum e membros de quórum foram adicionadas;
- O tipo das regras LLD rules foram alteradas de "Database monitor" para "Dependent item";
- itens com chave db.odbc.discovery foram transformados para itens dependentes do item db.odbc.get.

CHANGES IN 6.0.27

New templates

A new template is available:

- [YugabyteDB by HTTP](#), which includes the YugabyteDB Cluster by HTTP template for monitoring each YugabyteDB cluster.

CHANGES IN 6.0.28

New templates

A new template is available:

- [MSSQL by Zabbix agent 2](#)

CHANGES IN 6.0.29

Updated templates

- [FortiGate by SNMP](#) template has been supplemented with metrics regarding VPN, high availability (HA), wireless termination points (WTPs), SD-WAN health checks, and HW sensors.

CHANGES IN 6.0.30

New templates

The AWS ELB template set has been supplemented with the template [AWS ELB Network Load Balancer by HTTP](#).

8 Problemas conhecidos

Consulte também: [Compilation issues](#).

Inicialização do Proxy com MySQL 8.0.0-8.0.17

O zabbix_proxy em versões do MySQL 8.0.0-8.0.17 falha com o seguinte erro de "acesso negado":

```
[Z3001] connection to database 'zabbix' failed: [1227] Access denied; you need (at least one of) the SUPER
```

Isso ocorre devido ao MySQL 8.0.0 começar a aplicar permissões especiais para definir variáveis de sessão. No entanto, na versão 8.0.18 o comportamento foi removido: [A partir do MySQL 8.0.18, definir o valor desta variável do sistema não é mais uma operação restrita](#).

A solução alternativa é baseada na concessão de privilégios adicionais ao usuário zabbix :

Para versões do MySQL 8.0.14 - 8.0.17:

```
grant SESSION_VARIABLES_ADMIN on *.* to 'zabbix'@'localhost';
```

Para versões do MySQL 8.0.0 - 8.0.13:


```
grant SYSTEM_VARIABLES_ADMIN on *.* to 'zabbix'@'localhost';
```

Timescale DB: alto uso de memória com grande número de partições

As versões do PostgreSQL de 9.6 a 12 consomem muita memória ao atualizar tabelas com um grande número de partições [consulte o relatório do problema](#)). Este problema se manifesta quando o Zabbix atualiza tendências em sistemas com TimescaleDB se as tendências forem divididas em pedaços relativamente pequenos (por exemplo, 1 dia). Isso leva a centenas de pedaços presentes nas tabelas de tendências com as configurações de limpeza padrão - a condição em que o PostgreSQL provavelmente ficará sem memória.

O problema foi resolvido desde o Zabbix 5.0.1 para novas instalações com TimescaleDB, mas se TimescaleDB foi configurado com Zabbix antes disso, consulte [ZBX-16347](#) para notas de imigração.

Timescale DB 2.5.0: a política de compressão pode falhar em tabelas que contêm números inteiros

Este problema se manifesta quando o TimescaleDB 2.5.0 é usado. Foi resolvido desde o TimescaleDB 2.5.1.

Para mais informações, consulte [TimescaleDB Issue #3773](#).

Atualização

Configuração do modo SQL para uma configuração bem-sucedida

A configuração `sql_mode` no MySQL/MariaDB deve ter o modo "STRICT_TRANS_TABLES" definido. Se estiver ausente, a atualização do banco de dados do Zabbix falhará (consulte também [ZBX-19435](#)).

Atualização com MariaDB 10.2.1 e versões anteriores

A atualização do Zabbix pode falhar se as tabelas do banco de dados forem criadas com o MariaDB 10.2.1 e anteriores, pois nessas versões o formato padrão de linha é compacto. Isso pode ser corrigido alterando o formato de linha para dinâmico (consulte também [ZBX-17690](#)).

Instalação acidental de pacotes do Zabbix do EPEL

Com o repositório EPEL instalado e habilitado, a instalação do Zabbix a partir de pacotes levará à instalação dos pacotes do Zabbix do EPEL em vez dos pacotes oficiais do Zabbix.

Neste caso, desinstale os pacotes do Zabbix do EPEL, por exemplo:

```
dnf remove zabbix-server-mysql
```

Bloqueie os pacotes do Zabbix do EPEL. Adicione a seguinte linha no arquivo `/etc/yum.conf` :

```
exclude=zabbix6.0*
```

Instale o servidor do Zabbix novamente:

```
dnf install zabbix-server-mysql
```

Observe que os pacotes oficiais do Zabbix têm a palavra `release` na sua string de versão.

```
6.0.25-release1.el8
```

Zabbix packages for RHEL on Red Hat UBI environments

When installing Zabbix from Red Hat Enterprise Linux packages on [Red Hat Universal Base Image](#) environments, ensure access to required repositories and dependencies. Zabbix packages depend on `libOpenIPMI.so` and `libOpenIPMIposix.so` libraries, which are not provided by any package in the default package manager repositories enabled on UBI systems and will result in installation failures.

The `libOpenIPMI.so` and `libOpenIPMIposix.so` libraries are available in the `OpenIPMI-libs` package, which is provided by the `redhat-#-for-<arch>-appstream-rpms` repository. Access to this repository is curated by subscriptions, which, in the case of UBI environments, get propagated by mounting repository configuration and secrets directories of the RHEL host into the container file-system namespace.

For more information, see [ZBX-24291](#).

Conexão TLS com o banco de dados usando MariaDB

A conexão TLS com o banco de dados não é suportada com a opção `verify_ca` option para o parâmetro `parameter` `DBTLSConnect` se o MariaDB não foi utilizado.

Possíveis deadlocks com MySQL/MariaDB

Ao executar sob carga elevada e com mais de um trabalhador LLD envolvido, é possível encontrar um deadlock causado por um erro do InnoDB relacionado à estratégia de bloqueio de linhas (consulte [upstream bug](#)). O erro foi corrigido no MySQL desde a versão 8.0.29, mas não no MariaDB. Para mais detalhes, consulte [ZBX-21506](#).

Correlação global de eventos

Os eventos podem não ser correlacionados corretamente se o intervalo de tempo entre o primeiro e o segundo evento for muito pequeno, ou seja, meio segundo ou menos.

Intervalo do tipo Numeric (float) com PostgreSQL 11 ou anterior

PostgreSQL 11 e versões anteriores apenas suportam valores de ponto flutuante no intervalo aproximado de -1.34E-154 a 1.34E+154.

NetBSD 8.0 e mais recente

Vários processos do Zabbix podem quebrar aleatoriamente na inicialização nas versões NetBSD 8.X e 9.X. Isto se deve ao valor padrão de stack ser muito pequeno (4MB), o qual deve ser aumentado executando:

```
ulimit -s 10240
```

Para mais informações, por favor consulte o relatório de problema associado: [ZBX-18275](#).

Limitações de expressão regular no Zabbix Agent 2

O Zabbix agent 2 não suporta lookaheads e lookbehinds em expressões regulares devido às limitações da biblioteca padrão Go regexp.

Verificações IPMI

As verificações IPMI não funcionarão com o pacote de biblioteca padrão OpenIPMI no Debian anterior a 9 (stretch) e Ubuntu anterior a 16.04 (xenial). Para corrigir isto, recompile a biblioteca OpenIPMI com OpenSSL habilitado conforme discutido em [ZBX-6139](#).

Verificações SSH

- Algumas distribuições Linux como Debian, Ubuntu, não suportam chaves privadas criptografadas (com senha) se a biblioteca libssh2 for instalada a partir de pacotes. Para mais detalhes, consulte [ZBX-4850](#).
- Ao usar o libssh 0.9.x em algumas distribuições Linux com OpenSSH 8, verificações SSH podem ocasionalmente relatar "Não é possível ler dados do servidor SSH". Isso é causado por um problema no libssh [issue](#) ([relatório mais detalhado](#)). O erro deverá ter sido corrigido por uma versão estável do libssh 0.9.5. Para mais detalhes, consulte também [ZBX-17756](#).
- O uso do caractere pipe "|" no script SSH pode levar a um erro "Não é possível ler dados do servidor SSH". Nesse caso, é recomendável atualizar a versão da biblioteca libssh. Para mais detalhes, consulte também [ZBX-21337](#).

Verificações ODBC

- O driver MySQL unixODBC não deve ser usado com servidor Zabbix ou proxy Zabbix compilados com biblioteca de conector MariaDB e vice-versa. Se possível, é melhor evitar usar o mesmo conector que o driver devido um [bug de upstream](#). Configuração sugerida:

PostgreSQL, SQLite ou Oracle connector → MariaDB ou MySQL unixODBC driver

MariaDB connector → MariaDB unixODBC driver

MySQL connector → MySQL unixODBC driver

Para mais informações e correções disponíveis, consulte [ZBX-7665](#).

- Dados XML consultados a partir de Microsoft SQL Server podem ficar truncados de várias formas nos sistemas Linux and UNIX.
- Tem sido observado que o uso de verificações ODBC no CentOS 8 para monitoramento de banco de dados Oracle utilizando Oracle Instant Client para Linux 11.2.0.4.0 provoca quebra do Zabbix Server. Este problema pode ser resolvido pela atualização do Oracle Instant Client para 12.1.0.2.0, 12.2.0.1.0, 18.5.0.0.0 ou 19. Para mais informações, consulte [ZBX-18402](#).

Parâmetro do método de solicitação incorreto em itens

O parâmetro do método de solicitação, usado apenas em verificações HTTP, pode ser definido incorretamente como '1', um valor não padrão para todos os itens como resultado da atualização de uma versão do Zabbix anterior à 4.0. Para mais detalhes sobre como corrigir, consulte [ZBX-19308](#).

Monitoramento Web e agente HTTP

O servidor do Zabbix causa vazamento de memória (memory leak) no CentOS 6, CentOS 7 e possivelmente outras distribuições Linux relacionadas devido a um [bug de upstream](#) quando a opção "SSL verify peer" está habilitada nos cenários web ou agente HTTP. Para mais informações e correções disponíveis, consulte [ZBX-10486](#).

Verificações simples

Há um bug nas versões de **fping** anteriores a v3.10 que gerencia incorretamente pacote de resposta duplicados. Isto pode causar resultados inesperados para itens `icmping`, `icmppingloss` e `icmppingsec`. É recomendado utilizar a versão mais recente do **fping** disponível. Para mais detalhes, consulte [ZBX-11726](#).

Erros com a execução do `fping` em contêineres sem raiz

Quando os contêineres estão sendo executados no modo sem raiz ou em um ambiente com restrições específicas, você pode enfrentar erros relacionados à execução do `fping` ao realizar verificações ICMP, como `fping: Operation not permitted` ou todos os pacotes para todos os recursos perdidos.

Para corrigir esse problema, adicione `--cap-add=net_raw` aos comandos `"docker run"` ou `"podman run"`.

Além disso, a execução do `fping` em ambientes não raiz pode exigir modificação do `sysctl`, por exemplo:

```
sudo sysctl -w "net.ipv4.ping_group_range=0 1995"
```

onde "1995" é o GID do `zabbix`. Para mais detalhes, consulte [ZBX-22833](#).

Verificações SNMP

Se o sistema operacional OpenBSD for utilizado, um bug de `use-after-free` na biblioteca `Net-SNMP` até a versão 5.7.3 pode causar uma falha no servidor Zabbix se o parâmetro `SourceIP` for definido no arquivo de configuração do servidor Zabbix. Como solução alternativa, por favor, não defina o parâmetro `SourceIP`. O mesmo problema se aplica também ao Linux, mas não causa a interrupção do funcionamento do servidor Zabbix. Um patch local para o pacote `net-snmp` no OpenBSD foi aplicado e será lançado com o OpenBSD 6.3.

Picos nos dados SNMP

Picos nos dados SNMP têm sido observados e podem estar relacionados a certos fatores físicos como picos de tensão na rede. Para mais detalhes, consulte [ZBX-14318](#).

SNMP traps

O pacote `"net-snmp-perl"`, necessário para SNMP traps, foi removido no RHEL/CentOS 8.0-8.2; readicionado no RHEL 8.3.

Então se você estiver usando o RHEL 8.0-8.2, a melhor solução é atualizar para RHEL 8.3.

Para mais informações, consulte [ZBX-17192](#).

Falha no processo de alerta no RHEL 7

Ocorrências de falhas no processo de alerta do servidor Zabbix foram encontradas no RHEL 7. Para mais detalhes, consulte [ZBX-10461](#).

Atualizado o Zabbix agent 2 (6.0.5 ou anterior)

Ao atualizar o Zabbix agent 2 (versão 6.0.5 ou anterior) a partir de pacotes, pode ocorrer um erro de conflito de arquivo relacionado ao plugin. Para corrigir o erro, faça o backup da sua configuração do agente 2 (se necessário), desinstale o agente 2 e instale-o novamente.

Em sistemas baseados no On RHEL, execute:

```
dnf remove zabbix-agent2
dnf install zabbix-agent2
```

Em sistemas baseados no Debian-based, execute:

```
apt remove zabbix-agent2
apt install zabbix-agent2
```

Para mais informações, consulte [ZBX-23250](#).

Invertendo as localidades do frontend

Observou-se que as localidades do frontend podem ser invertidas sem lógica aparente, ou seja, algumas páginas (ou partes das páginas) são exibidas em um idioma enquanto outras páginas (ou partes das páginas) em um idioma diferente. Tipicamente, o problema pode aparecer quando há vários usuários, alguns dos quais usam uma localidade, enquanto outros usam outra.

Uma solução conhecida para isso é desativar a `multithreading` no PHP e no Apache.

O problema está relacionado com a forma como a configuração da localidade funciona em PHP [in PHP](#): as informações de localidade são mantidas por processo, não por thread. Assim, em um ambiente multi-thread, quando vários projetos são executados pelo mesmo processo do Apache, é possível que a localidade seja alterada em outro thread e que isso altere como os dados podem ser processados no thread do Zabbix.

Para mais informações, consulte os relatórios de problemas relacionados:

- [ZBX-10911](#) (Problema com inversão das localidades do frontend)
- [ZBX-16297](#) (Problema com processamento de números em gráficos usando a função `bcdiv` das funções BC Math)

Configuração opcache no PHP 7.3

Se "opcache" está habilitado na configuração do PHP 7.3, o Zabbix Frontend pode mostrar uma tela em branco quando carregado pela primeira vez. Este é um [bug do PHP](#) registrado. Para contornar este problema, por favor configure o parâmetro "opcache.optimization_level" para 0x7FFFBFDF no arquivo de configuração do PHP (arquivo `php.ini`).

Gráficos

Horário de verão

Alterações no Horário de Verão resultam em irregularidades ao exibir rótulos do eixo X (duplicação de datas, datas faltando, etc.).

Agregação de Soma

Ao usar agregação de soma **sum aggregation** em um gráfico para um período inferior a uma hora, os gráficos exibem valores incorretos (multiplicados) quando os dados vêm de tendências.

Sobreposição de Texto

Para alguns idiomas do frontend (por exemplo, japonês), as fontes locais podem causar sobreposição de texto na legenda do gráfico. Para evitar este problema, use a versão 2.3.0 (ou posterior) da extensão PHP GD.

Monitoramento de arquivo de log

Os itens `log[]` e `logrt[]` releem repetidamente o arquivo de log desde o início se o sistema de arquivos estiver 100% ocupado e o arquivo de log estiver sendo anexado (Para mais informações, consulte [ZBX-10884](#).)

Consultas lentas do MySQL

O servidor Zabbix gera consultas `SELECT queries in case of non-existing values for items`. Este problema [issue](#) é conhecido por ocorrer nas versões MySQL 5.6/5.7 (para uma discussão mais detalhada, consulte [ZBX-10652](#)), e em casos específicos, também pode ocorrer em versões mais recentes do MySQL. Uma solução alternativa para isso é desativar o otimizador `index_condition_pushdown` ou `prefer_ordering_index` no MySQL. No entanto, observe que esta solução alternativa pode não corrigir todos os problemas relacionados a consultas lentas.

Sincronização lenta de configuração com Oracle

A sincronização de configuração pode ser lenta em instalações do Zabbix 6.0 com Oracle DB que possuem um grande número de itens e etapas de pré-processamento de itens. Isso é causado pela velocidade de processamento do motor de banco de dados Oracle para campos do tipo `nclob`.

Para melhorar o desempenho, você pode converter os tipos de campo de `nclob` para `nvarchar2` aplicando manualmente o patch do banco de dados `items_nvarchar_prepare.sql`. Observe que essa conversão reduzirá o limite máximo de tamanho do campo de 65535 bytes para 4000 bytes para parâmetros de pré-processamento de itens e parâmetros de itens, como `Description`, campo `Script` do item do script, campos `Request body` e `Headers` do item do agente HTTP, campo consulta SQL `query` do item do monitor de banco de dados. Consultas para determinar os nomes dos modelos que precisam ser excluídos antes de aplicar o patch são fornecidas no patch como um comentário. Alternativamente, se `MAX_STRING_SIZE` estiver configurado, você pode alterar `nvarchar2(4000)` para `nvarchar2(32767)` nas consultas do patch para definir o limite de tamanho do campo de 32767 bytes.

Para uma discussão mais detalhada, consulte [ZBX-22363](#).

Login na API

Um grande número de sessões de usuário abertas pode ser criado ao utilizar scripts personalizados com o método `user.login` **method** sem `user.logout`.

Problemas no IPv6 em armadilhas SNMPv3

Devido a um bug no `net-snmp`, o endereço IPv6 pode não ser exibido corretamente ao usar SNMPv3 em armadilhas SNMP. Para mais detalhes e uma possível solução alternativa, consulte [ZBX-14541](#).

Endereço IPv6 longo truncado em informações de login com falhas

Uma mensagem de tentativa de login falhada exibirá apenas os primeiros 39 caracteres de um endereço de IP armazenado, pois esse é o limite de caracteres no campo do banco de dados. Isso significa que endereços de IP IPv6 com mais de 39 caracteres serão exibidos de forma incompleta.

Verificações do Zabbix Agent no Windows

Valores de DNS inexistentes configurados no parâmetro `Server` do arquivo de configuração do Zabbix Agent (`zabbix_agentd.conf`) podem elevar o tempo de resposta do Zabbix Agent no Windows. Isso ocorre porque o serviço de cache de DNS do Windows não

armazena respostas negativas para endereços IPv4. No entanto, para endereços IPv6 as respostas negativas são armazenadas, então uma possível contingência para este problema seja desabilitar o protocolo IPv4 na máquina.

Exportação/Importação YAML

Existem alguns problemas conhecidos com a YAML **export/import**:

- Mensagens de erro não são traduzíveis;
- SON válido com uma extensão de arquivo .yaml às vezes não pode ser importado;
- Datas legíveis por humanos não citadas são automaticamente convertidas para carimbos de data/hora Unix.

Assistente de instalação no SUSE com NGINX e php-fpm

O assistente de instalação do Frontend não pode salvar o arquivo de configuração no SUSE com NGINX + php-fpm. Isto é causado por uma configuração no serviço `/usr/lib/systemd/system/php-fpm.service`, que impede o Zabbix de gravar em `/etc`. (introduzido no [PHP 7.4](#)).

Existem duas opções de solução alternativa disponíveis:

- Configure a opção **ProtectSystem** para 'true' em vez de 'full' na unidade systemd do php-fpm.
- Salve manualmente o arquivo `/etc/zabbix/web/zabbix.conf.php`.

Chromium para serviço web do Zabbix no Ubuntu 20

Embora na maioria dos casos, o serviço web do Zabbix possa ser executado com o Chromium, no Ubuntu 20.04 o uso do Chromium causa o seguinte erro:

```
Cannot fetch data: chrome failed to start:cmd_run.go:994:
WARNING: cannot create user data directory: cannot create
"/var/lib/zabbix/snap/chromium/1564": mkdir /var/lib/zabbix: permission denied
Sorry, home directories outside of /home are not currently supported. See https://forum.snapcraft.io/t/112
```

Esse erro ocorre porque `/var/lib/zabbix` é usado como diretório home do usuário 'zabbix'.

Códigos de erros personalizados do MySQL

Se o Zabbix for utilizado com uma instalação do MySQL no Azure, uma mensagem de erro pouco clara [9002] Alguns erros ocorreram pode aparecer nos logs do Zabbix. Este texto genérico de erro é enviado para o servidor ou proxy do Zabbix pelo banco de dados. Para obter mais informações sobre a causa do erro, verifique os logs do Azure.

Expressões regulares inválidas após a troca para PCRE2

No Zabbix 6.0, o suporte para PCRE2 foi adicionado. Embora o PCRE ainda seja suportado, os pacotes de instalação do Zabbix para RHEL 7 e mais recentes, SLES (todas as versões), Debian 9 e mais recentes, Ubuntu 16.04 e mais recentes foram atualizados para usar o PCRE2. Embora forneça muitos benefícios, a mudança para o PCRE2 pode fazer com que certos padrões de expressões regulares PCRE existentes se tornem inválidos ou se comportem de maneira diferente. Em particular, isso afeta o padrão `^[\\w\\.]`. Para tornar essa expressão regular válida novamente sem afetar a semântica, mude a expressão para `^[\\w\\.]`. Isso acontece devido ao fato de que o PCRE2 trata o sinal de traço como um delimitador, criando uma faixa dentro de uma classe de caracteres. Os seguintes pacotes de instalação do Zabbix foram atualizados e agora usam PCRE2: RHEL 7 e mais recentes, SLES (todas as versões), Debian 9 e mais recentes, Ubuntu 16.04 e mais recentes.

Conversão incorreta de serviços no Zabbix 6.0.0-6.0.2

Na versão 6.0 do Zabbix, foram introduzidos novos algoritmos de cálculo de status de serviço mais flexíveis.

Após uma atualização do Zabbix <6.0 para o Zabbix 6.0.0, 6.0.1, 6.0.2, as regras de cálculo de status de serviço 'Mais crítico se todos os filhos tiverem problemas' e 'Mais crítico dos serviços filhos' serão trocadas. Os serviços criados no Zabbix 6.0.0 e posterior terão regras corretas de cálculo de status.

Ao atualizar de versões <6.0 para o Zabbix 6.0.3 ou posterior, o Zabbix atualizará corretamente as regras de cálculo de status de serviço. A atualização de 6.0.x para 6.0.3 não terá efeito nas regras de cálculo de status de serviço.

Erro no widget Geomap

Os mapas no widget Geomap podem não carregar corretamente se você tiver feito upgrade de uma versão mais antiga do Zabbix com o NGINX e não tiver mudado para o novo arquivo de configuração do NGINX durante a atualização.

Para corrigir o problema, você pode descartar o arquivo de configuração antigo, usar o arquivo de configuração do pacote 6.0 e reconfigurá-lo conforme descrito nas instruções (https://www.zabbix.com/download?zabbix=6.0&os_distribution=red_hat_enterprise_linux&os_instructions) na seção e. Configure PHP para Zabbix frontend.

Alternativamente, você pode editar manualmente um arquivo de configuração existente do NGINX (tipicamente, `/etc/zabbix/nginx.conf`). Para fazer isso, abra o arquivo e localize o seguinte bloco:

```
location ~ /(api\|/|conf[^\.]|include|locale|vendor) {
    deny          all;
    return        404;
}
```

Then, replace this block with:

```
location ~ /(api\|/|conf[^\.]|include|locale) {
    deny          all;
    return        404;
}

location /vendor {
    deny          all;
    return        404;
}
```

Logrotate para servidor e proxy do Zabbix

O utilitário logrotate está incluído apenas nos pacotes para zabbix-agent, zabbix-agent2 e zabbix-web-service, mas precisa ser instalado separadamente para o servidor e o proxy do Zabbix. A dependência do logrotate foi adicionada aos pacotes do servidor e do proxy para RHEL e SUSE a partir do Zabbix 6.4.4rc1.

Problemas no Zabbix 6.0.11

Erros de análise de JSONPath

Erros de análise de JSONPath ocorrem em caso de espaços em branco iniciais e array/object vazio. Esses erros foram corrigidos no Zabbix 6.0.12.

Avaliação de E/OU em filtros LLD

A avaliação de expressões AND/OR nos filtros/sobreposições de descoberta de baixo nível pode falhar nesta versão. Corrigido no Zabbix 6.0.12.

Arquivos ausentes no arquivo do agente do Windows

O arquivo ZIP de download do agente Zabbix para Windows está sem os arquivos zabbix_sender.h e zabbix_sender.lib nas versões 6.0.0-6.0.27, necessários para o zabbix zabbix_sender.dll.

1 Problemas de compilação

Estes são os problemas conhecidos relacionados à compilação do Zabbix a partir das fontes. Para todos os outros casos, consulte a página [Known issues](#).

Compilando o agente Zabbix no HP-UX

Se você instalar a biblioteca PCRE a partir do site popular de pacote HP-UX <http://hpux.connect.org.uk> (por exemplo, a partir do arquivo pcre-8.42-ia64_64-11.31.depot), apenas a versão 64-bit da biblioteca será instalada no diretório /usr/local/lib/hpux64.

Nesse caso, para a compilação bem-sucedida do agente, uma opção personalizada é necessária para o script configure, por exemplo:

```
CFLAGS="+DD64" ./configure --enable-agent --with-libpcre-include=/usr/local/include --with-libpcre-lib=/usr
```

Biblioteca em um local não padrão

O zabbix permite especificar uma biblioteca localizada em uma local não padrão. No exemplo abaixo, o Zabbix executará curl-config a partir do local não padrão especificado e usará sua saída para determinar a versão correta da libcurl a ser utilizada.

```
$ ./configure --enable-server --with-mysql --with-libcurl=/usr/local/bin/curl-config
```

Isso funcionará se for a única instalação da liburl no sistema, mas pode não funcionar se houver outra libcurl instalada em um local padrão (por exemplo, pelo gerenciador de pacotes). Esse é o caso quando você precisa de uma versão mais recente da biblioteca para o Zabbix e a versão antiga para outras aplicações.

Portanto, especificar um componente em um local não padrão nem sempre funcionará quando o mesmo componente também existir em um local padrão.

Por exemplo, se você utilizar uma versão mais recente da libcurl instalada /usr/local, o Zabbix pode selecionar a versão errada e a compilação falhar:

```
usr/bin/ld: ../../src/libs/zbxhttp/libzbxhttp.a(http.o): in function 'zbx_http_convert_to_utf8':  
/tmp/zabbix-master/src/libs/zbxhttp/http.c:957: undefined reference to 'curl_easy_header'  
collect2: error: ld returned 1 exit status
```

Aqui, a função `curl_easy_header()` não está disponível na versão mais antiga `/usr/lib/x86_64-linux-gnu/libcurl.so`, mas está disponível na versão mais recente `/usr/local/lib/libcurl.so`.

O problema está na ordem das flags do linker, e uma solução é especificar o caminho completo para a biblioteca em uma variável `LD_FLAGS`:

```
$ LD_FLAGS="-Wl,--no-as-needed /usr/local/lib/libcurl.so" ./configure --enable-server --with-mysql --with-l
```

Observe que a opção `-Wl,--no-as-needed` pode ser necessária para alguns sistemas (consulte também: opções padrão de linking nos sistemas [Debian-based](#)).

10 Notas de atualização para 6.0.0

Estas notas são para atualização do Zabbix 5.4.x para o Zabbix 6.0.0. Todas as notas estão agrupadas em:

- **Crítico** - as informações mais críticas relacionadas ao processo de atualização e às mudanças na funcionalidade do Zabbix
- **Informação** - todas as informações restantes descrevendo as mudanças na funcionalidade do Zabbix

É possível atualizar para o Zabbix 6.0.0 a partir de versões do Zabbix anteriores à 5.4.0. Consulte a seção [procedimento de atualização](#) para todas as informações relevantes sobre atualização de versões anteriores do Zabbix.

Crítico Bancos de dados

Para criar uma melhor experiência de usuário e garantir a melhor performance ao Zabbix em vários ambientes de produção, o suporte a algumas versões mais antigas de banco de dados foi removido. Isto se aplica primariamente às versões de banco de dados que estão próximas do fim do seu ciclo de vida e versões com problemas não corrigidos que podem interferir com a performance normal.

A partir do Zabbix 6.0, as seguintes versões de [banco de dados](#) são oficialmente suportadas:

- MySQL/Percona 8.0.X
- MariaDB 10.5.X - 10.6.X
- PostgreSQL 13.X
- Oracle 19c - 21c
- TimescaleDB 2.0.1-2.3
- SQLite 3.3.5-3.34.X

Por padrão, o Zabbix Server e Proxy não irão iniciar se uma versão de banco de dados não suportada for detectada. Agora é possível, no entanto não recomendado, desativar a verificação de versão de BD modificando o parâmetro de configuração `AllowUnsupportedDBVersions` para o [Server](#) ou [Proxy](#).

Chaves primárias

As chaves primárias são agora usadas para todas as tabelas, incluindo as tabelas históricas, em novas instalações.

Não há atualização automática para chaves primárias para instalações existentes. Instruções para uma [atualização manual](#) das tabelas históricas para chaves primárias em instalações pré-existentes estão disponíveis para [MySQL/MariaDB](#), [PostgreSQL](#), [TimescaleDB v1](#) e [v2](#), e [Oracle](#).

Suporte PCRE2

O suporte a PCRE2 foi adicionado. PCRE ainda é suportado, mas o Zabbix só pode ser compilado com uma das bibliotecas PCRE ou PCRE2, ambas não podem ser usadas ao mesmo tempo.

Os seguintes pacotes de instalação do Zabbix foram atualizados e agora usam PCRE2: - RHEL/CentOS 7 e mais recentes - SLES (todas as versões) - Debian 9 e mais recentes - Ubuntu 16.04 e mais recentes

Por favor note que após a alteração para PCRE2, você pode precisar atualizar algumas expressões regulares. Em particular, o padrão `^[\w-\.]` precisa ser alterado para `^-[\w\\.\.]` para continuar funcionando corretamente - veja [problemas conhecidos](#) para uma explicação mais detalhada.

Processamento separado para verificações ODBC

O processamento de verificações ODBC agora são executadas por processos Server/Proxy separados `odbc pollers`. Anteriormente, as verificações ODBC eram executadas por `pollers` regulares, que também operam com itens do Zabbix Agent, verificações SSH, etc.

Um novo parâmetro de configuração StartODBCPollers foi adicionado aos arquivos de configuração do Zabbix **Server** e **Proxy** com o valor padrão 1. Este parâmetro pode necessitar de ajuste baseado no número de verificações ODBC executadas pelo Server ou Proxy. Você pode também querer reduzir o número de pollers regulares definido pelo parâmetro StartPollers apropriadamente.

O item interno zabbix[process,<type>] pode ser usado para monitorar a carga de pollers ODBC.

Log de auditoria

De forma a melhorar o log de auditoria no Zabbix e torná-lo completo e confiável, a estrutura de banco de dados anteriormente existente precisou ser retrabalhada. Durante uma atualização as tabelas auditlog e auditlog_details serão substituídas pela nova tabela auditlog com um formato diferente. **Registro de auditoria antigos não serão preservados.**

Uma nova **seção** Log de auditoria foi adicionada ao menu Administração → Geral permitindo habilitar (padrão) ou desabilitar o log de auditoria. As definições de limpeza (housekeeping) para auditoria, previamente localizada sob a seção de menu Housekeeper, também foi movida à nova seção Log de auditoria. As definições de limpeza existentes serão salvas.

Mudanças na API

Consulte a lista de **mudanças na API** no Zabbix 6.0.0.

Macros simples substituídas por macros de expressão

A funcionalidade das macros simples foi **transferida** para as macros de expressão. As macros simples existentes serão convertidas para macros de expressão durante a atualização. Macros que não podem ser convertidas sem exceder o limite de comprimento não serão convertidas com um alerta registrado no arquivo de log.

Macros

Macros posicionais não mais suportadas

O suporte às macros posicionais no nome do item (\$1, \$2...\$9), obsoletas desde o Zabbix 4.0, foram completamente removidas.

Macros de usuário no nome do item não mais suportadas

O suporte para macros de usuário nos nomes do item (incluindo nomes de regras de descoberta), obsoleto desde o Zabbix 4.0, foi completamente removido.

Monitoramento → Visão geral removido

A seção Visão geral no menu Monitoramento foi removido completamente. A mesma funcionalidade pode ser ainda acessada pelo uso dos **widgets** de dashboard Visão geral de dados e Visão geral de gatilho.

Alteração de dependência para gatilhos herdados desabilitada

A possibilidade de alterar as dependências para gatilhos herdados de um modelo está agora desabilitada. A razão é que através da atualização das dependências de um gatilho modelo, as dependências dos gatilhos herdados são sobrescritas. Portanto é mais confiável sempre definir as dependências de gatilho apenas no nível do modelo raiz.

Informativo Itens internos obsoletos para histórico/tendências

Os seguintes itens internos estão agora obsoletos e serão removidos em uma versão primária (major) futura:

- zabbix[history]
- zabbix[history_log]
- zabbix[history_str]
- zabbix[history_text]
- zabbix[history_uint]
- zabbix[trends]
- zabbix[trends_uint]

Plugins do Zabbix Agent 2

Cada plugin do Zabbix Agent 2 agora possui um **arquivo de configuração** separado. Por padrão, estes arquivos estão localizados no diretório ./zabbix_agent2.d/plugins.d/. O caminho é especificado no parâmetro Include do arquivo de configuração do Agent 2 e pode ser relativo à localização do arquivo **zabbix_agent2.conf** ou **zabbix_agent2.win.conf**.

Senhas de usuário

Anteriormente, os espaços nas senhas de usuários eram automaticamente cortados em ambos o formulário de configuração do Usuário e no formulário de Autenticação. Após a introdução de **requisitos de complexidade de senha** configuráveis, os espaços na senha não são mais cortados. Então usuários que acham que tenham espaços em suas senhas, não poderão se autenticar de forma usual e precisarão informar sua senha 'antiga' sem espaços. Para continuar a usar senhas com espaços, eles terão que recriar suas senhas.

Processamento em lote para métricas Prometheus

Como o processamento em lote de itens dependentes foi introduzido na fila de pré-processamento para as métricas Prometheus, itens dependentes não serão mais processados em paralelo e podem ter suas velocidades de processamento afetadas.

Transferência de comando de tempo de execução

Os comando de tempo de execução do Zabbix Server e Proxy são agora enviados via socket em vez de sinais Unix. Esta mudança permite melhorar a experiência de usuário operando com opções de controle de tempo de execução:

- Os resultados da execução do comando são agora exibidos no console.
- É possível enviar parâmetros de entrada mais longos, tais como nome de nó HA em vez do número do nó.

Gráficos customizados favoritos não mais suportados

Não é mais possível adicionar gráficos customizados aos favoritos em Monitoramento -> Hosts -> Gráficos. Após a atualização quaisquer gráficos customizados existentes serão removidos dos favoritos.

Monitoramento de serviço

Várias **major updates** relacionadas à funcionalidade de monitoramento de serviço foram feitas. Uma configuração de árvore de serviço existente será alterada durante um upgrade da seguinte maneira:

- Dependências baseadas em Triggers entre problemas e serviços são substituídas por mapeamento baseado em tags de serviços para problemas. Triggers que foram vinculadas a um serviço receberão uma nova tag `ServiceLink`: `<trigger ID>:<nome da trigger>` (o valor da tag será truncado para 32 caracteres). Os serviços vinculados receberão a mesma **problem tag**.
- Dependências hard e soft não existem mais. Em vez disso, um serviço terá vários serviços pai.
- O 'algoritmo de cálculo de status' será atualizado usando as seguintes regras: -- Não calcular → Definir status para OK -- Problema, se pelo menos um filho tiver um problema → Mais crítico dos nós filhos -- Problema, se todas as crianças tiverem problemas → Mais crítico se todas as crianças tiverem problemas
- SLA não é mais um atributo de serviço, mas uma entidade separada que pode ser atribuída a vários serviços. Durante uma atualização, SLAs idênticos serão agrupados e um SLA por cada grupo será criado. Os serviços receberão uma nova **service tag** `SLA:<ID>` para correspondência.

Veja também: -- Descrição detalhada de **upgrade de monitoramento de serviço**; - Configuração de **serviços**.

11 Notas de atualização para 6.0.1

Configuração sincronizada

A performance da configuração sincronizada foi melhorada. É recomendável melhorar o CacheSize no server/proxy se houver uma grande quantidade de templates.

Também é recomendado remover os templates não usados.

Observe que o CacheSize padrão no server/proxy foi melhorado para 32M.

Mudanças no Item

Suporte nativo para os **items net.dns** e **net.dns.record** foi adicionado ao agente Zabbix 2. No agente Zabbix 2 para Windows, agora esses itens permitem endereços de IP DNS personalizados no parâmetro `ip` e não mais ignoram os parâmetros de `timeout` e `count`.

12 Notas de atualização para 6.0.2

Esta versão menor não possui notas de atualização.

13 Notas de atualização para 6.0.3

Esta versão menor não possui notas de atualização.

14 Notas de atualização para 6.0.4

Esta versão menor não possui notas de atualização.

15 Notas de atualização para 6.0.5

Esta versão menor não possui notas de atualização.

16 Notas de atualização para 6.0.6

Plugin MongoDB carregável O plugin MongoDB [plugin](#) não faz mais parte do Zabbix agente 2 e agora está disponível como um plugin carregável. A lista de versões suportadas do MongoDB foi estendida para 2.6-5.3.

A funcionalidade do plugin e o conjunto de itens suportados [items](#) não mudaram.

O plugin MongoDB, junto com outros plugins carregáveis que podem ser adicionados no futuro, está armazenado em um novo repositório [zabbix-agent2-plugins](#). Agora o pacote zabbix-release adiciona esse repositório ao sistema do usuário.

Para continuar monitorando Mongo DB nas instalações a partir dos pacotes Zabbix oficial, atualizar o pacote zabbix-release e instalar o pacote zabbix-agent2-plugin-mongodb.

Fontes estão disponíveis em [CDN](#).

17 Notas de atualização para 6.0.7

Expansão do nome Symlink Agora o nome do Symlink e o caminho completo do symlink retornam nos itens `vfs.dir.get[]` and `vfs.file.get[]`, em vez de serem resolvidos no target symlink.

18 Notas de atualização para 6.0.8

Esta versão menor não possui notas de atualização.

19 Notas de atualização para 6.0.9

Esta versão menor não possui notas de atualização.

20 Notas atualizadas para 6.0.10

Mudanças incompatíveis

Plugin PostgreSQL movido para plugins carregáveis O plugin PostgreSQL [plugin](#) não está mais incorporado no agente Zabbix 2. Agora ele é um plugin carregável no agente 2.

Esta mudança pode/irá quebrar as automações com Ansible, Chef, etc, porque não é mais possível baixar o repositório de plugin diretamente. Ver também: [PostgreSQL loadable plugin](#) repositório

21 Notas atualizações para 6.0.11

Mudanças incompatíveis

Atualização DB schema Uma nova coluna `name_upper` foi adicionado a table **items** da base de dados. A coluna contém uma versão em maiúsculas do nome do item ou do nome da regra LLD. Essa alteração foi implementada para permitir o uso do índice com pesquisa com maiúsculas e minúsculas para consultas de API **optimization**.

Consultas de API otimizadas As consultas de bancos de dados API, utilizadas ao pesquisar por nomes nas tabelas `hosts` e `items`, foram otimizadas e agora serão processadas de forma mais eficiente. Como resultado dessa alteração, triggers determinístico precisam ser criados durante uma atualização.

No MySQL e MariaDB, é necessário que `GLOBAL log_bin_trust_function_creators = 1` seja definido se o logging binário estiver ativado e não houver privilégios de superuser, e `log_bin_trust_function_creators = 1` não estiver configurado no arquivo de configuração MySQL. Para configurar a variável utilizando o console MySQL, execute:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
```

Uma vez que a atualização foi completada com sucesso, o `log_bin_trust_function_creators` pode ser desabilitado:

```
mysql> SET GLOBAL log_bin_trust_function_creators = 0;
```

Triggers também são criados para bancos de dados PostgreSQL e Oracle.

Erros de análise do JSONPath Os erros de análise de JSONPath ocorrem nessa versão em casos de espaços em branco iniciais e em casos de arrays/objetos vazios. Corrido em Zabbix 6.0.12.

Avaliação AND/OR nos filtros de LLD A avaliação de expressões AND/OR nos filtros/overrides de descoberta baixo nível pode falhar nesta versão. Corrigido em Zabbix 6.0.12.

22 Upgrade notas para 6.0.12

Melhoria de desempenho nos sincronizadores de histórico O desempenho de sincronizadores de histórico foi aprimorado, introduzindo uma nova trava de leitura e escrita. Isso reduz o travamento entre os sincronizadores de histórico, trappers e proxy poller, usando uma trava de leitura compartilhada ao acessar o cache de configuração. A nova trava pode ser bloqueada somente para escrita pelo sincronizador de configuração que realiza a recarga do cache de configuração.

23 Notas de atualização para 6.0.13

Mudanças incompatíveis

 Versionamento de plugin carregável

Agora, para **Loadable plugins** o agente Zabbix 2 utiliza o mesmo versionamento de sistema como no próprio Zabbix. As mudanças abaixo foram realizadas:

- MongoDB 1.2.0 -> MongoDB 6.0.13
- PostgreSQL 1.2.1 -> PostgreSQL 6.0.13

Esses plugins são suportados por qualquer versão menor do Zabbix 6.0. Perceba que a fonte do repositório de código para cada plugin agora contém uma nova branch `release/6.0` (anteriormente, havia apenas `omaster branch`).

Configuration import Ver mudanças nos processos **configuration import**.

Consultar tablespaces separados em bancos de dados Oracle com agente Zabbix 2 O **Zabbix agent 2 items** suportados pelo plugin Oracle, agora tem os parâmetros opcionais abaixo:

- `oracle.diskgroups.stats[<existingParameters>,<diskgroup>]`
- `oracle.archive.info[<existingParameters>,<destination>]`
- `oracle.cdb.info[<existingParameters>,<database>]`

- oracle.pdb.info[<existingParameters>,<database>]
- oracle.ts.stats[<existingParameters>,<tablespace>,<type>]

Esses parâmetros permitem consultar instâncias separadas de dados em vez de todos os dados, assim melhorando o desempenho.

Mudanças nos itens

Suporte nativo para os itens **net.dns** e **net.dns.record** foi adicionado ao agente Zabbix 2. No agente Zabbix 2 para Windows, esse item agora permite endereços IP DNS personalizados no parâmetro `ip` e não ignoram os parâmetros `timeout` e `count`.

24 Notas de upgrade para 6.0.14

Limits for JavaScript objects in preprocessing The following limits for **JavaScript objects** in preprocessing have been introduced:

- The total size of all messages that can be logged with the `Log()` method has been limited to 8 MB per script execution.
- The initialization of multiple `HttpRequest` objects has been limited to 10 per script execution.
- The total length of header fields that can be added to a single `HttpRequest` object with the `addHeader()` method has been limited to 128 Kbytes (special characters and header names included).

25 Notas de atualização para 6.0.15

Esta versão menor não possui notas atualizadas.

26 Notas de atualização para 6.0.16

Esta versão menor não possui notas de atualização.

27 Notas sobre a atualização para 6.0.17

Suporte HTML removido na atribuição do Geomap O texto atribuído para **Geomap dashboard widget** agora pode somente conter texto puro; o suporte ao HTML foi removido. Se este campo já conter HTML, será renderizado como texto puro depois da atualização.

Nas configurações do **Geographical maps** em Administração → Seção Geral, o campo Atribuição agora fica somente visível quando Tile provider é configurado para Outro.

28 Notas de atualização para 6.0.18

Manutenção de histórico de proxy A limitação na quantidade de informações desatualizadas excluídas do banco de dados do proxy por ciclo de manutenção do histórico de proxy foi removida. Anteriormente, o **housekeeper** excluía não mais que 4 vezes as horas **HousekeepingFrequency** de informações desatualizadas. Por exemplo, se **HousekeepingFrequency** estivesse configurado como "1", não era excluído mais do que 4 horas de informações desatualizadas (iniciando a partir da entrada mais antiga). Nos casos em que um proxy constantemente recebesse dados mais antigos do que definido em **ProxyOfflineBuffer**, isso poderia resultar em um acúmulo excessivo de dados. Agora essa limitação foi removida, proporcionando uma solução de gerenciamento de proxy mais eficaz.

29 Upgrade notas para 6.0.19

Funções de agregação A função **count_foreach** agora retorna para um item correspondente no array, caso não haja dados presentes para o item ou caso não haja dados que correspondam ao filtro. Anteriormente, esses itens eram ignorados. ignored (nenhum dado adicionado à agregação).

30 Notas de atualização para 6.0.20

Esta versão menor não possui notas de atualização.

31 Notas de atualização para 6.0.21

Profundidade máxima do JSON

A profundidade máxima permitida do JSON foi definida como 64 para melhorar a segurança e performance durante a análise do JSON.

Funções macro

O intervalo do `fmtnum` macro `function` agora está limitada a 0-20.

32 Notas de atualização para 6.0.22

Tabela de autoregistro limpa de registros órfãos.

Anteriormente a tabela `autoreg_host` nunca era limpada. Agora, registros órfãos que não fazem referência a um evento de autoregistro nem a um host existente serão removidos pelo Housekeeper.

Parâmetros do plugin PostgreSQL

Os parâmetros do plugin PostgreSQL não são mais obrigatórios se os `Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect` estiverem configurados como `verify_ca` ou `verify_full`:

- `Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile`
- `Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile`

33 Notas de atualização para 6.0.23

Mudanças incompatíveis Plugins agente 2

Agora o master host correto retorna se for especificado `mysql.replication.get_slave_status[connString,<username>,<password>,<item>]` no parâmetro "masterHost". Anteriormente, o parâmetro "masterHost" era ignorado e sempre retornava o master host primeiro. Se o parâmetro não for especificado, todos os hosts retornam.

Parâmetros do plugin MySQL

Os parâmetros do plugin MySQL plugin não são mais obrigatórios se os `Plugins.Mysql.Sessions.<SessionName>.TLSConnect` estiverem configurados como `verify_ca` ou `verify_full`:

- `Plugins.Mysql.Sessions.<SessionName>.TLSCertFile`
- `Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile`

A combinação PostgreSQL-TimescaleDB não é mais verificada

O Zabbix não verifica mais a combinação PostgreSQL-TimescaleDB suportada. Como antes before, o Zabbix verifica a versão suportada do PostgreSQL ou a versão TimescaleDB separadamente.

34 Notas de atualização 6.0.24

Esta versão menor não possui notas de atualização.

Parâmetros do plugin MongoDB

Os seguintes parâmetros do plugin MongoDB não são mais obrigatórios se `Plugins.MongoDB.Sessions.<SessionName>.TLSConnect` estiver configurado para `verify_ca` ou `verify_full`:

- `Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile`
- `Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile`

Ver também: [MongoDB plugin parameters](#)

35 Notas de atualização para 6.0.25

Consistência introduzida nos checksums sha256 do item `vfs.file.cksum`

Nas versões anteriores do Zabbix, o item Zabbix agent `vfs.file.cksum` produzia somas sha256 diferentes para o mesmo arquivo, dependendo da plataforma (arquitetura do processador) – as somas sha256 em AIX, HP-UX (Itanium) e Solaris (SPARC) são diferentes das produzidas em plataformas Intel x86-64. Esse problema foi resolvido. Contudo, após a atualização, monitorar as somas sha256 de arquivos em AIX, HP-UX ou Solaris SPARC pode resultar em falsos positivos de arquivos que foram modificados.

36 Notas de atualização para 6.0.26

Esta versão menor não tem notas de atualização.

37 Notas de atualização para 6.0.27

Suporte ao Zabbix agent 2 support no Windows

Para evitar vulnerabilidades de segurança críticas, a versão mínima do Windows para o Zabbix Agent 2 foi elevada para Windows 10/Server 2016. Para mais informações, consulte [Supported platforms](#).

38 Notas atualizadas para 6.0.28

Autorização de usuário convidado

O login automático para o usuário convidado foi removido. Após essa alteração, o usuário convidado precisará fazer login como qualquer outro usuário. Anteriormente, um convidado poderia acessar quase qualquer página de monitoramento ou relatório sem passar pela autorização.

Expressão regular inválida em itens `proc.*`

Itens de agente **`proc.*`** agora se tornarão 'não suportados' se uma expressão regular inválida for fornecida.

39 Atualizar notas para 6.0.29

Esta versão menor não possui nota de atualização.

39 Upgrade notes for 6.0.30

This minor version doesn't have any upgrade notes.

5. Início rápido

Por favor utilize a barra lateral para acessar o conteúdo na seção de Início rápido.

1 Autenticando e configurando usuário

Visão geral

Nesta seção você aprenderá como autenticar e configurar um usuário de sistema no Zabbix.

Autenticação

ZABBIX

Username

Password

☒ Remember me for 30 days

Sign in

or sign in as guest

Esta é a tela de boas-vindas do Zabbix. Informe o nome de usuário **Admin** com a senha **zabbix** para se autenticar como um **super usuário do Zabbix**. O acesso aos menus Configuração e Administração estarão garantidos.

Proteção contra ataques de força bruta

No caso de cinco tentativas falhas de autenticação consecutivas, o Zabbix pausará por 30 segundos para prevenir ataques de força bruta e de dicionário.

O endereço IP de uma tentativa de autenticação falha será exibido após uma autenticação bem-sucedida.

Adicionando usuário

Para visualizar informações sobre usuários, vá até Administração → Usuários.

<input type="checkbox"/> Alias	Name	Surname	User role	Groups	Is online?	Login	Frontend access	API access	Debug mode	Status
<input type="checkbox"/> Admin	Zabbix	Administrator	Super admin role	Zabbix administrators	Yes (2020-10-28 11:38:16)	OK	System default	Enabled	Enabled	Enabled
<input type="checkbox"/> guest	John	Snow	User role	Guests	No (2020-07-16 11:06:52)	OK	System default	Enabled	Disabled	Disabled

Displaying 2 of 2 found

Para adicionar um novo usuário, clique em Criar usuário.

No formulário de novo usuário, certifique-se de adicionar seu usuário em um dos **grupos de usuário** existentes, por exemplo 'Administradores Zabbix'.

User
Media
Permissions

* Alias
user

Name
New

Surname
User

* Groups
Zabbix administrators X
type here to search

* Password

* Password (once again)

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Por padrão, novos usuários não possuem mídia (métodos de entrega de notificação) definida para eles. Para criar uma, vá até a aba 'Mídia' e clique em Adicionar.

Media

Type
Email

* Send to
user@domain.tld
Remove

Add

* When active
1-7,00:00-24:00

Use if severity
☒ Not classified
☒ Information
☒ Warning
☒ Average
☒ High
☒ Disaster

Enabled
☒

Add

Cancel

Neste tela, informe um endereço de e-mail para o usuário.

Você pode especificar um período de tempo quanto a mídia estará ativa (consulte a página [Especificação de período de tempo](#) para uma descrição do formato), por padrão uma mídia está sempre ativa. Você também pode customizar os níveis de [severidade de gatilho](#) para os quais a mídia estará ativa, mas deixe todos habilitados por enquanto.

Clique em Adicionar para salvar a mídia, então vá até a aba Permissões.

A aba Permissões tem um campo obrigatório Função (role). A função determina quais elementos do Frontend o usuário pode

UserMediaPermissions

• Role

Admin role ✕

Select

User type

Admin

Permissions

Host group

All groups

Permissions

None

Permissions can be assigned for user groups only.

Access to UI elements

Monitoring

DashboardProblemsHostsOverviewLatest dataMapsDiscoveryServices

Inventory

OverviewHosts

Reports

Availability reportTriggers top 100NotificationsScheduled reports

Configuration

Host groupsTemplatesHostsMaintenanceActionsDiscoveryServices

Access to modules

No enabled modules found.

Access to API

Enabled

Access to actions

Create and edit dashboardsCreate and edit mapsCreate and edit maintenance

Add problem commentsChange severityAcknowledge problemsClose problemsExecute scripts

Manage API tokensManage scheduled reports

Add

Cancel

<input type="checkbox"/>	Alias	Name	Surname	User role	Groups	Is online?	Login	Frontend access	API access	Debug mode	Status
<input type="checkbox"/>	Admin	Zabbix	Administrator	Super admin role	Zabbix administrators	Yes (2020-10-28 11:42:05)	OK	System default	Enabled	Enabled	Enabled
<input type="checkbox"/>	guest	John	Snow	User role	Guests	No (2020-07-16 11:06:52)	OK	System default	Enabled	Disabled	Disabled
<input type="checkbox"/>	user			Admin role	Zabbix administrators	No	OK	System default	Enabled	Enabled	Enabled

Displaying 3 of 3 found

Por padrão, um novo usuário não tem permissões de acesso a hosts. Para fornecer tais acessos, clique no grupo do usuário na coluna Grupos (neste caso - 'Administradores Zabbix'). No formulário de propriedades do grupo, vá até a aba Permissões.

≡ User groups

User groupPermissionsTag filter

Permissions

Host group

All groups

Permissions

None

type here to search

Select

Read-wr

☐ Include subgroups

Add

Update

Delete

Cancel

Este usuário deve ter acesso somente leitura ao grupo Linux servers, então clique em Selecionar próximo ao campo de seleção de grupo do usuário.

Host groups

☐ Name

☐ Discovered hosts

☐ Hypervisors

☒ Linux servers

☐ Templates

☐ Templates/Applications

☐ Virtual machines

☐ Zabbix servers

Select

Neste tela, marque a caixa de seleção próximo a 'Linux servers', então clique em Selecionar. Linux servers deve ser exibido no campo de seleção. Clique no botão 'Leitura' para definir o nível de permissão e então Adicionar para adicionar o grupo à lista de permissões. No formulário de propriedades do grupo do usuário, clique em Atualizar.

Attention:

No Zabbix, os direitos de acesso aos hosts são atribuídos para os **grupos de usuário**, não para os usuários de forma individual.

Pronto! Você pode tentar autenticar usando as credenciais do novo usuário.

2 Novo host

Visão geral

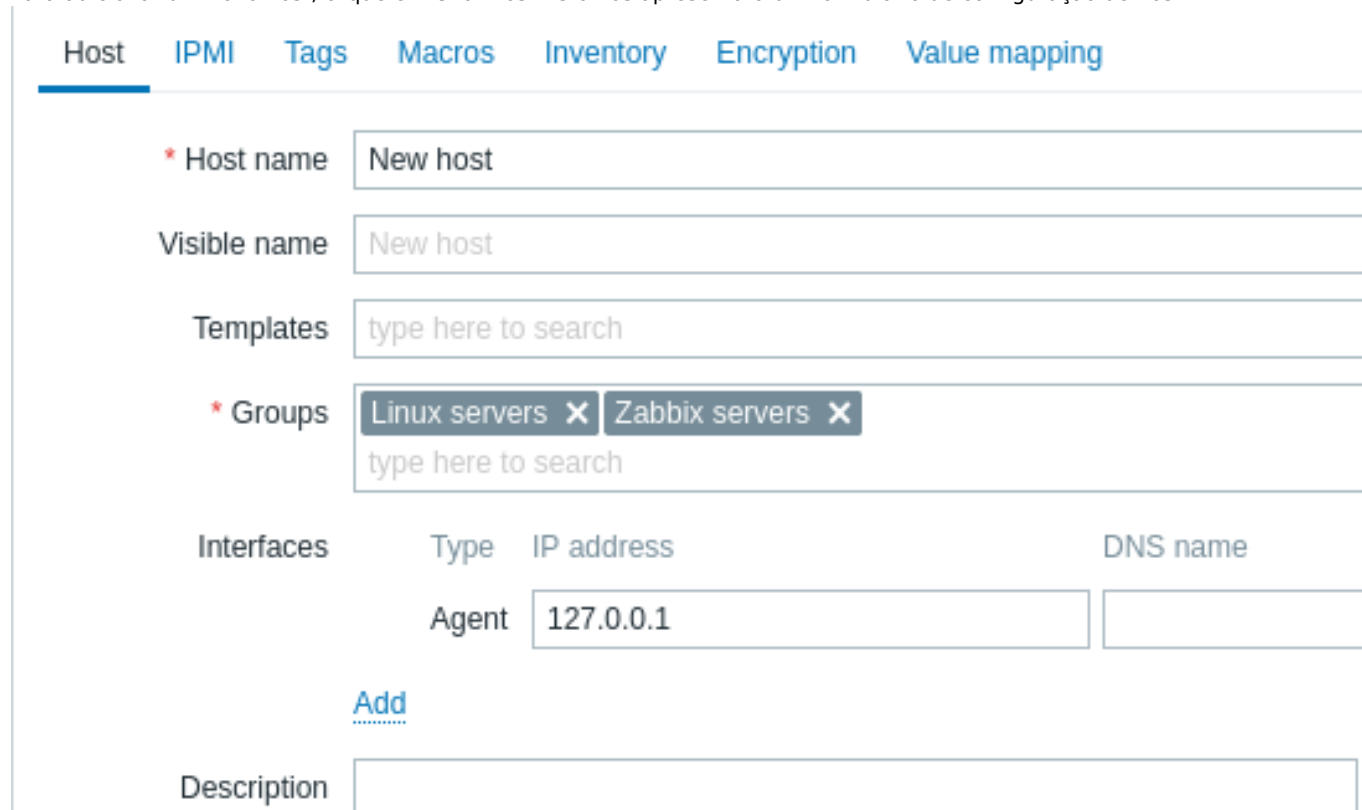
Nesta seção você aprenderá a como configurar um novo host.

Um host no Zabbix é uma entidade de rede (física, virtual) que você deseja monitorar. A definição do que pode ser um "host" no Zabbix é bastante flexível. Ele pode ser um servidor físico, um switch de rede, uma máquina virtual ou alguma aplicação.

Adicionando um host

Informações sobre hosts configurados no Zabbix estão disponíveis em Configuração → Hosts e Monitoramento → Hosts. Há um host já pré-definido, chamado "Servidor Zabbix" (Zabbix server), mas queremos aprender como adicionar um outro.

Para adicionar um novo host, clique em Criar host. Isto nos apresentará um formulário de configuração de host.



Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

O mínimo a ser informado aqui é:

Nome do host

- Informe um nome de host. Caracteres alfanuméricos, espaços, pontos, traços e subtraços são permitidos.

Grupos

- Selecione um ou vários grupos existentes clicando no botão Selecionar ou informar um grupo inexistente para criar um novo grupo.

Note:

Todas as permissões de acesso são associadas aos grupos de host, não aos hosts individualmente. Por isso um host deve pertencer a um grupo pelo menos.

Interfaces: endereço IP

- Apesar de tecnicamente não ser um campo obrigatório, você pode querer informar o endereço IP do host. Note que se este é o endereço IP do Zabbix Server, ele deve ser especificado na diretiva 'Server' do arquivo de configuração do Zabbix Agent.

Outras opções nos atenderão com seus valores padrão por agora.

Quando pronto, clique em Adicionar. Seu novo host deve estar visível na lista de hosts.

Hosts


Create host



Import

Filter

<input type="checkbox"/>	Name ▲	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availability	Agent encryption	Info	Tags
<input type="checkbox"/>	New host	Applications	Items	Triggers	Graphs	Discovery	Web	127.0.0.1: 10050		New template	Enabled	ZBX SNMP JMX IPMI	NONE		

A coluna Disponibilidade contém indicadores de disponibilidade do host para cada interface. Nós definimos uma interface de Zabbix Agent, então podemos usar o ícone de disponibilidade do agente (com 'ZBX' nele) para entender a disponibilidade do host:

-  - estado do host não foi estabelecido; nenhuma verificação de métrica ocorreu ainda

-  - host está disponível, uma verificação métrica foi bem-sucedida
-  - host está indisponível, uma verificação métrica falhou (mova seu cursor do mouse sobre o ícone para ver a mensagem de erro). Pode existir algum erro de comunicação, possivelmente causado por credenciais de interface incorretas. Certifique-se de que Zabbix Server esteja em execução, e tente recarregar a página também após isso.

3 Novo item

Visão geral

Nesta seção, você aprenderá como configurar um item.

Itens são a base da coleta de dados no Zabbix. Sem os itens, não há dados - porque apenas um item define uma métrica ou que tipo de dado coletar de um host.

Adicionando item

Todos os itens são agrupados em torno dos hosts. Por isso é que para configurar uma amostra de item nós vamos até Configuração → Hosts e procuramos pelo "Novo host" que acabamos de criar.

Clique no link Itens na linha do "Novo host", e então clique em Criar item. Isto nos apresentará um formulário de definição de item.

Item
Tags
Preprocessing

* Name

CPU load

Type

Zabbix agent

* Key

system.cpu.load

Type of information

Numeric (float)

* Host interface

127.0.0.1:10050

Units

* Update interval

1m

Custom intervals

Type	Interval	Period
Flexible	Scheduling	50s
Add		

* History storage period

Do not keep history

Storage period

90d

* Trend storage period

Do not keep trends

Storage period

365d

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Para nossa amostra de item, as informações essenciais a serem definidas são:

Nome

- Informe Carga CPU como valor. Este será o nome do item exibido nas listas e em outros lugares.

Chave

- Manualmente informe system.cpu.load como valor. Este é o nome técnico de um item que identifica o tipo de informação que será coletado. A chave em particular é apenas uma das **chaves pré-definidas** que vêm com o Zabbix Agent.

Tipo de informação

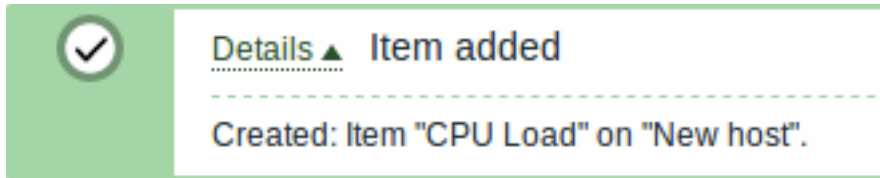
- Este atributo define o formato do dado esperado. Para a chave `system.cpu.load`, este campo será automaticamente definido para Numérico (float).

Note:

Você pode também desejar diminuir o número de dias em que o **histórico do item** será mantido, para 7 ou 14. Esta é uma boa prática para aliviar o banco de dados de manter uma grande quantidade de valores históricos.

Outras opções nos atenderão com seus valores padrão por agora.

Quando pronto, clique em Adicionar. O novo item deve aparecer na lista de item. Clique em Detalhes sobre a lista para visualizar o que foi feito exatamente.



Visualizando os dados

Com um item definido, você deve estar curioso se ele está de fato coletando dados. Para tal, vá para Monitoramento → Últimos dados, selecione o host 'Novo host' no filtro e clique em Aplicar.

Latest data

Host	Name	Last check	Last value	Change	Tags
New host	CPU load	05/24/2021 10:40:5...	1.17	-0.11	

0 selected Display stacked graph Display graph

Com isto dito, levará até 60 segundos para o primeiro dado chegar. Esta, por padrão, é a frequência com que o Server lê as alterações de configuração e capta novos itens para execução.

Se você não ver qualquer valor na coluna 'Alteração', talvez apenas um valor tenha sido recebido até o momento. Aguarde 30 segundos para a chegada de outro valor.

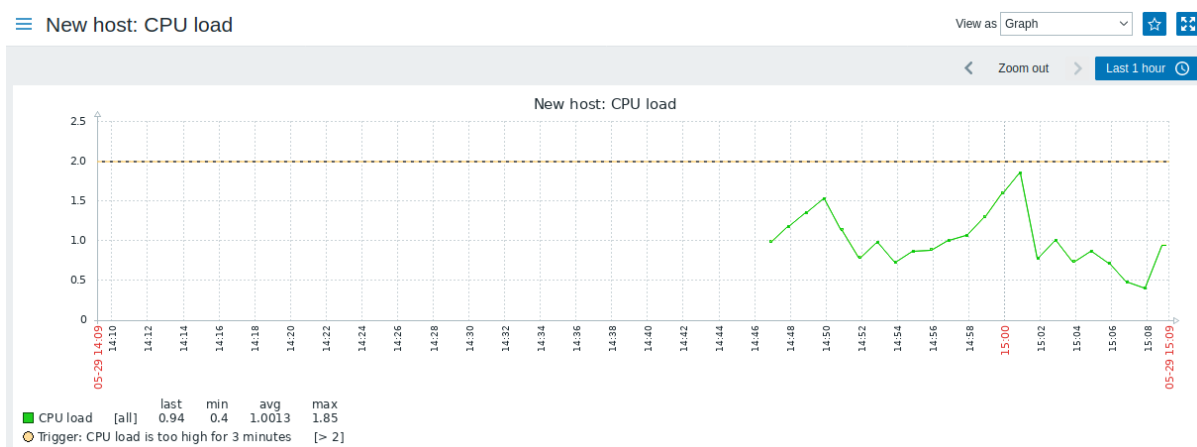
Se você não vê informações sobre o item como na imagem, certifique-se de que:

- você preencheu os campos 'Chave' e 'Tipo de informação' do item exatamente como na imagem
- ambos Agent e Server estão em execução
- estado do host é 'Monitorado' e seu ícone de disponibilidade está verde
- um host está selecionado na lista de host, o item está ativo

Gráficos

Com o item funcionando por um tempo, deve ser tempo de visualizar alguma coisa visual. **Gráficos simples** estão disponíveis para qualquer item numérico monitorado sem qualquer configuração adicional. Estes gráficos são gerados em tempo de execução.

Para visualizar o gráfico, vá até Monitoramento → Últimos dados e clique no link 'Gráfico' próximo ao item.



4 Novo gatilho (trigger)

Visão geral

Nesta seção você aprenderá como configurar um gatilho.

Itens apenas coletam dados. Para avaliar automaticamente os dados entrantes nós precisamos definir gatilhos. Um gatilho contém uma expressão que define um limite do que é um nível aceitável para o dado.

Se este nível é ultrapassado pelo dado de entrada, um gatilho irá “disparar” ou entrará em um estado de ‘Problema’ - permitindo que tomemos consciência de que algo aconteceu e que pode necessitar de atenção. Se o nível voltar a ser aceitável, o gatilho retorna para um estado ‘Ok’.

Adicionando um gatilho

Para configurar um gatilho para nosso item, vá até Configuração → Hosts, encontre o ‘Novo host’ e clique em Gatilhos próximo a ele e então em Criar gatilho. Isto nos apresenta um formulário de definição de gatilho.

Trigger

Tags

Dependencies

*

Name

CPU load too high on 'New host' for 3 minutes

Event name

CPU load too high on 'New host' for 3 minutes

Operational data

Severity

Not classified

Information

Warning

Average

High

Dis

*

Expression

avg(/New host/system.cpu.load, 3m)>2

Add

[Expression constructor](#)

OK event generation

Expression

Recovery expression

None

PROBLEM event generation mode

Single

Multiple

OK event closes

All problems

All problems if tag values match

Allow manual close

☐

URL

Description

Enabled

☒

Add

Cancel

Para nosso gatilho, as informações essenciais são:

Nome

- Informe Carga de CPU muito alta no ‘Novo host’ por 3 minutos como valor. Este será o nome do gatilho exibido nas lista e em outros lugares.

Expressão

- Informe: `avg(/Novo host/system.cpu.load,3m)>2`

Esta é a expressão do gatilho. Certifique-se de que a expressão seja gravada corretamente, até o último símbolo. A chave de item aqui (`system.cpu.load`) é usada para se referir ao item. Esta expressão em particular basicamente diz que o limite (threshold) de problema é excedido quando o valor médio de carga de CPU por 3 minutos estiver acima de 2. Você pode aprender mais sobre a [sintaxe de expressões de gatilho](#).

Quando pronto, clique em Adicionar. O novo gatilho deve aparecer na lista de gatilhos.

Exibindo estado do gatilho

Com um gatilho definido, você deve estar interessado em ver seu estado.

Se a carga de CPU excedeu o nível limite que você definiu no gatilho, o problema será exibido em Monitoramento → Problemas.

Time	<input type="checkbox"/> Severity	Recovery time	Status	Info	Host ▲	Problem	Operational data	Duration
16:23:06	<input type="checkbox"/> Not classified		PROBLEM		New host	CPU load too high on "New host" for 3 minutes	6.6	56s

A intermitência na coluna de estado indica uma mudança recente do estado do gatilho, uma que tenha surgido nos últimos 30 minutos.

5 Recebendo notificação de problema

Visão geral

Nesta seção você aprenderá como configurar alertas na forma de notificação no Zabbix.

Com itens coletando dados e gatilhos projetados para "disparar" em situações de problema, seria também útil ter algum mecanismo de alerta disponível pelo qual seríamos notificados sobre eventos importantes mesmo quando não estivermos olhando diretamente para o Zabbix Frontend.

Isto é o que fazem as notificações. Sendo o e-mail o método de entrega mais popular para notificações de problema, nós aprenderemos como configurar uma notificação por e-mail.

Definições de e-mail

Inicialmente há vários **métodos de entrega** de notificações pré-definidos no Zabbix. **E-mail** é um deles.

Para configurar as definições de e-mail, vá até Administração → Tipos de mídia e clique em Email na lista de tipos de mídia pré-definidos.

Media types

<input type="checkbox"/>	Name ▲	Type	Status	Used in actions	Details
<input type="checkbox"/>	Email	Email	Enabled		SMTP server: "mail.zabbix.com",
<input type="checkbox"/>	Mattermost	Webhook	Enabled		
<input type="checkbox"/>	Opsgenie	Webhook	Enabled		

Isto nos apresentará um formulário para definição das configurações de e-mail.

Media types

Media type	Message templates	Options
		<div><div>* Name</div><div>Email</div></div>
		<div><div>Type</div><div>Email</div></div>
		<div><div>* SMTP server</div><div>mail.zabbix.com</div></div>
		<div><div>SMTP server port</div><div>25</div></div>
		<div><div>* SMTP helo</div><div>zabbix.com</div></div>
		<div><div>* SMTP email</div><div>zabbix-info@zabbix.com</div></div>
		<div><div>Connection security</div><div>None</div><div>STARTTLS</div><div>SSL/TLS</div></div>
		<div><div>Authentication</div><div>None</div><div>Username and password</div></div>
		<div><div>Message format</div><div>HTML</div><div>Plain text</div></div>
		<div><div>Description</div><div></div></div>
		<div><div>Enabled</div><div><input checked="" type="checkbox"/></div></div>
		<div><div>Add</div><div>Cancel</div></div>

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Defina os valores apropriados de servidor, helo e e-mail de SMTP, conforme seu ambiente.

Note:

'Email SMTP' será usado como endereço 'De' para as notificações enviadas pelo Zabbix.

Pressione Atualizar quando pronto.

Agora você configurou 'Email' como um tipo de mídia funcional. Um tipo de mídia deve ser associado aos usuários pela definição de endereços de entrega específicos (como fizemos quando [configurando um novo usuário](#)), caso contrário ele não será usado.

Nova ação

Entrega de notificações é uma das coisas que as [ações](#) fazem no Zabbix. Portanto, para configurar uma notificação, vá até Configuração → Ações e clique em Criar ação.

≡ Actions

Action

Operations

*

Name

Test action

Conditions

Label

Name

Add

Enabled

☒

*

At least one operation must exist.

Add

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Neste formulário, informe um nome para a ação.

No caso mais simples, se não adicionarmos nenhuma outra **condição** específica, a ação será levada em consideração em qualquer mudança de gatilho de 'Ok' para 'Problema'.

Nós ainda devemos definir o que a ação deve fazer - e isso é feito na aba Operações. Clique em Adicionar no bloco Operações, o que abrirá um novo formulário de operação.

Operation details

Operation type

Send message

Steps

1 - 1

(0 - infinitely)

Step duration

0

(0 - use action default)

* At least one user or user group must be selected.

Send to User groups

User group	Action
Add	

Send to Users

User	Action
user (New User)	Remove
Add	

Send only to

Email

Custom message

☐

Conditions

Label	Name	Action
Add		

Add

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Aqui, clique em Adicionar no bloco Enviar para Usuários e selecione o usuário ('user') que acabamos de definir. Selecione 'Email' como o valor de Enviar apenas para. Quando estiver feito, clique em Adicionar, e a operação deve ser adicionada:

≡ Actions

Action

Operations

* Default operation step duration

1h

Pause operations for suppressed problems

☒

Operations

Steps	Details	Start in	Duration
1	Send message to users: user (New User) via Email	Immediately	Default
Add			

Isto é tudo para a configuração de uma ação simples, então clique em Adicionar no formulário da ação.

Recebendo notificação

Agora, com a entrega de notificações configurada seria divertido realmente receber uma. Para ajudar neste ponto, podemos de propósito aumentar a carga em nosso host - de modo que nosso **gatilho** "dispare" e nós recebamos uma notificação de problema.

Abra o console no seu host e execute:

```
cat /dev/urandom | md5sum
```

Você pode executar um ou vários [desses processos](#).

Agora vá até Monitoramento → Últimos dados e veja como os valores de 'Carga de CPU' aumentaram. Lembre, para nosso gatilho disparar, o valor de 'Carga de CPU' deve passar de '2' por 3 minutos de execução. Assim que ocorrer:

- em Monitoramento → Problemas você deve ver o gatilho com um estado 'Problema' piscando
- você deve receber uma notificação de problema no seu e-mail

Attention:

Se a notificação não funcionar:

- Verifique mais uma vez se ambas as configurações de e-mail e da ação foram executadas corretamente
- Certifique-se de que o usuário criado por você possui ao menos permissão de leitura no host que gerou o evento, como citado na etapa **Adicionando usuário**. O usuário, sendo parte do grupo de usuários 'Administradores Zabbix' deve ter ao menos acesso de leitura ao grupo de hosts 'Servidores Linux' ao qual nosso host pertence.
- Adicionalmente, você pode verificar o log da ação indo em Relatórios → Log de ação.

6 Novo modelo (template)

Visão geral

Nesta seção você aprenderá como configurar um modelo.

Previamente nós aprendemos como configurar um item, um gatilho e como receber uma notificação de problema para o host.

Enquanto todas estas etapas oferecem uma boa flexibilidade por si mesmas, pode parecer como um monte de etapas a serem tomadas se necessário para, digamos, mil hosts. Alguma automação seria uma mão na roda.

Aqui é onde os modelos vêm nos socorrer. Modelos permitem agrupar itens, gatilhos e outras entidades úteis de forma que possam ser reutilizados continuamente pela aplicação aos hosts em uma única etapa.

Quando um modelo é associado a um host, o host herda todas as entidades do modelo. Então, basicamente um monte de verificações podem ser aplicadas muito rapidamente.

Adicionando modelo

Para começarmos a trabalhar com modelos, nós devemos primeiro criar um. Para fazermos isso, em Configuração → Modelos clique em Criar modelo. Isto nos apresentará um formulário de configuração de modelo.

The screenshot shows the 'Create Template' form in Zabbix. The 'Template' tab is selected. The form contains the following fields and controls:

- Template name**: A required field (marked with a red asterisk) containing the text 'New template'.
- Visible name**: A field containing the text 'New template'.
- Templates**: A field with a search prompt 'type here to search' and a 'Select' button.
- Groups**: A required field (marked with a red asterisk) containing 'Templates' with a close button (x) and a search prompt 'type here to search'. It also has a 'Select' button.
- Description**: A large text area for entering a description.
- Buttons**: 'Add' and 'Cancel' buttons at the bottom.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Os parâmetros requeridos aqui são:

Nome do modelo

- Informe um nome para o modelo. Caracteres alfanuméricos, espaços e subtraços são permitidos.

Grupos

- Selecione um ou vários grupos clicando no botão Selecionar. O modelo deve pertencer a um grupo.

Quando pronto, clique em Adicionar. Seu novo modelo deve estar visível na lista de modelos.

≡ Templates

<input type="checkbox"/>	Name ▲	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web
<input type="checkbox"/>	New template	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web

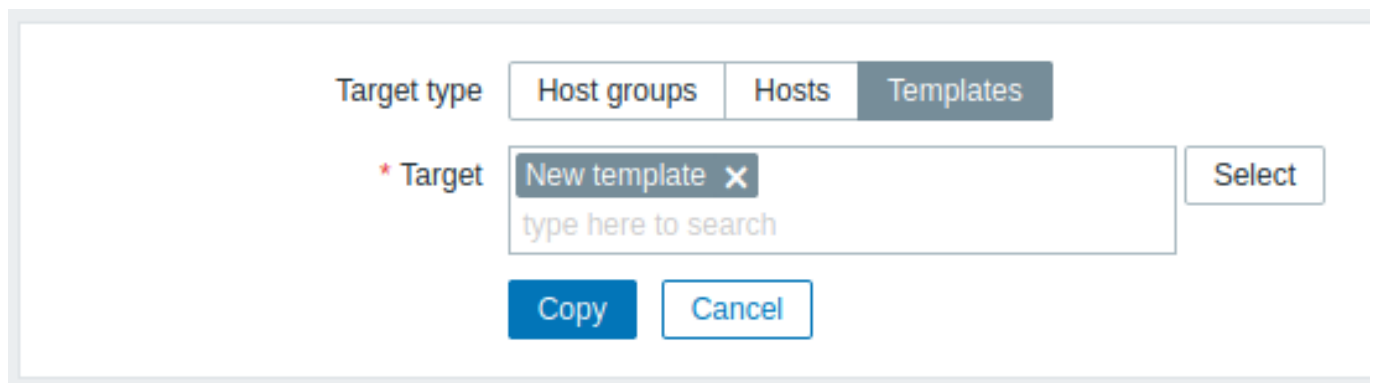
Como você pode ver, o modelo está lá, mas ele não tem nada nele - sem itens, gatilhos ou outras entidades.

Adicionando item ao modelo

Para adicionar um item ao modelo, vá até a lista de itens para o 'Novo host'. Em Configuração → Hosts clique em Itens próximo ao 'Novo host'.

Então:

- marque a caixa de seleção do item 'Carga de CPU' na lista
- clique em Copiar abaixo da lista
- selecione o modelo para o qual copiar o item



Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

- clique em Copiar

Se agora você for até Configuração → Modelos, o 'Novo modelo' deve ter um novo item nele.

Nós vamos parar em um item por agora, mas de forma similar você pode adicionar quaisquer outros itens, gatilhos e outras entidades ao modelo até que ele seja um conjunto razoavelmente completo de entidades para um dado propósito (monitoramento de SO, monitoramento de aplicação única).

Associando modelo ao host

Com um modelo pronto, resta apenas adicioná-lo a um host. Para isso, vá até Configuração → Hosts, clique em 'Novo host' para abrir seu formulário de propriedades e encontre o campo **Modelos**.

Comece digitando Novo modelo no campo Modelos. O nome do modelo que criamos deve aparecer na lista suspensa. Arraste para baixo para selecionar. Veja que ele aparece no campo Modelos.

Host

Host

IPMI

Tags

Macros

Inventory

Encryption

Value mapping

* Host name

New host

Visible name

New host

Templates

New template X

type here to search

* Groups

Linux servers X

type here to search

Clique em Atualizar no formulário para salvar as mudanças. O modelo está agora adicionado ao host, com todas as entidades que ele contém.

Como você deve ter suposto, desta forma ele pode ser aplicado a qualquer outro host. Quaisquer alterações nos itens, gatilhos e outras entidades no nível do modelo serão propagadas aos hosts aos quais o modelo está associado.

Associando modelos pré-definidos aos hosts

Como você deve ter notado, o Zabbix vem com um conjunto de modelo pré-definidos para vários SOs, dispositivos e aplicações. Para começar com o monitoramento de forma muito rápida, você pode associar o modelo apropriado a um host, mas cuide que estes modelos precisam estar bem ajustados para o seu ambiente. Algumas verificações podem não ser necessárias, e intervalos de sondagem (polling) podem ser frequentes demais.

Mais informações sobre [modelos](#) estão disponíveis.

6. Aplicação Zabbix

Visão geral Como alternativa a configurar manualmente ou reutilizar um servidor existente para o Zabbix, os usuários podem [baixar](#) uma aplicação Zabbix (Zabbix appliance) ou uma imagem de CD de instalação da aplicação Zabbix.

As versões da aplicação Zabbix e CD de instalação são baseadas em CentOS 8 (x86_64).

O CD de instalação da aplicação Zabbix pode ser usado para implantação imediata do Zabbix Server (MySQL).

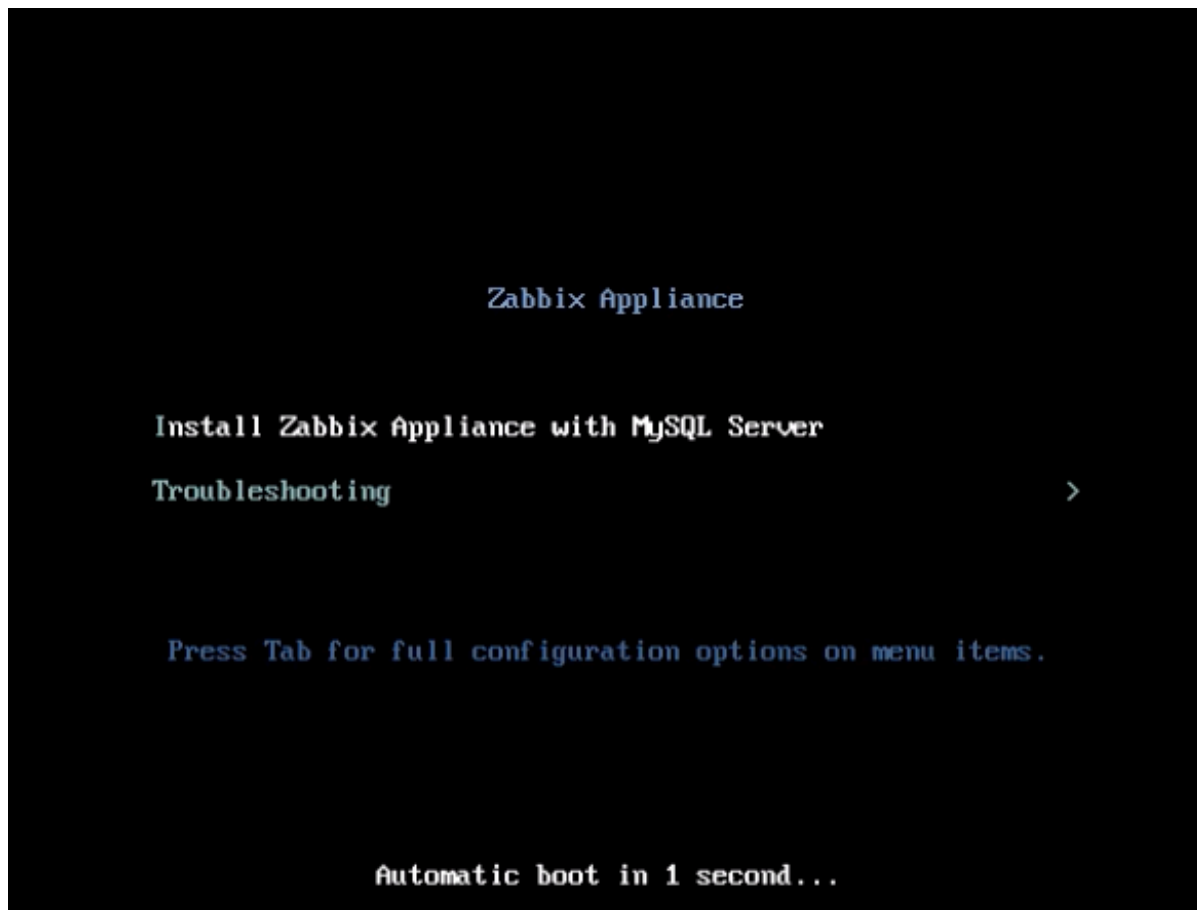
Attention:

Você pode usar esta aplicação para avaliar o Zabbix. A aplicação não objetiva o uso efetivo em produção.

Requisitos de Sistema:

- RAM: 1.5 GB
- Espaço em disco: no mínimo 8 GB devem ser alocados para a máquina virtual.

Menu de instalação do CD/DVD do Zabbix:



A aplicação Zabbix contém um Zabbix Server (configurado e rodando com MySQL) e um frontend.

A aplicação virtual do Zabbix está disponível nos seguintes formatos:

- VMWare (.vmx)
- Open virtualization format (.ovf)
- Microsoft Hyper-V 2012 (.vhdx)
- Microsoft Hyper-V 2008 (.vhd)
- KVM, Parallels, QEMU, USB stick, VirtualBox, Xen (.raw)
- KVM, QEMU (.qcow2)

Para iniciar, suba (boot) a aplicação e navegue até o endereço que a aplicação (VM) recebeu por DHCP.

Attention:

O DHCP deve estar habilitado na máquina virtualizadora.

Para verificar o endereço IP dentro da máquina virtual execute:

```
ip addr show
```

Para acessar o Frontend do Zabbix, navegue até **http://<host_ip>** (para acessar pelo navegador da máquina virtualizadora o modo bridged deve estar habilitado nas configurações de rede da máquina virtual).

Note:

Se a aplicação falhar ao iniciar no Hyper-V, você pode tentar pressionar Ctrl+Alt+F2 para alternar entre sessões tty.

1 Mudanças nas configurações do CentOS 8 A aplicação é baseada em CentOS 8. Há algumas mudanças aplicadas às configurações base do CentOS.

1.1 Repositórios

O **repositório** oficial do Zabbix foi adicionado ao /etc/yum.repos.d:

```
[zabbix]
name=Zabbix Official Repository - $basearch
baseurl=http://repo.zabbix.com/zabbix/5.2/rhel/8/$basearch/
enabled=1
```

```
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-ZABBIX-A14FE591
```

1.2 Configurações de firewall

A aplicação usa firewall iptables com regras pré-definidas:

- Porta SSH aberta (22 TCP);
- Portas Zabbix Agent (10050 TCP) e Zabbix Trapper (10051 TCP) abertas;
- Portas HTTP (80 TCP) e HTTPS (443 TCP) abertas;
- Portas SNMP trap aberta (162 UDP);
- Conexões de saída para porta NTP (53 UDP) abertas;
- Pacotes ICMP limitados a 5 pacotes por segundo;
- Todas as outras conexões de entrada são bloqueadas.

1.3 Usando um endereço IP estático

Por padrão a aplicação usa DHCP para obter um endereço IP. Para especificar um endereço IP estático:

- Acesse como usuário root;
- Abra o arquivo `/etc/sysconfig/network-scripts/ifcfg-eth0`;
- Altere `BOOTPROTO=dhcp` por `BOOTPROTO=none`
- Adicione as seguintes linhas:
 - `IPADDR=<IP address of the appliance>`
 - `PREFIX=<CIDR prefix>`
 - `GATEWAY=<gateway IP address>`
 - `DNS1=<DNS server IP address>`
- Execute o comando **`systemctl restart network`**.

Consulte a [documentação](#) oficial da Red Hat se necessário.

1.4 Alterando time zone

Por padrão a aplicação utiliza UTC para o relógio do sistema. Para alterar o time zone, copie o arquivo apropriado de `/usr/share/zoneinfo` para `/etc/localtime`, por exemplo:

```
cp /usr/share/zoneinfo/Europe/Riga /etc/localtime
```

2 Configuração Zabbix A configuração inicial da aplicação Zabbix tem as seguintes alterações de senha e configuração:

2.1 Credenciais (login:password)

Sistema:

- root:zabbix

Zabbix Frontend:

- Admin:zabbix

Banco de Dados:

- root:<random>
- zabbix:<random>

Note:

Senhas de banco de dados são geradas randomicamente durante o processo de instalação.
A senha de Root é arquivada no arquivo `/root/.my.cnf`. Não é necessário informar a senha ao usar a conta "root".

Para alterar a senha do usuário do banco de dados, as alterações devem ser feitas no seguintes locais:

- MySQL;
- `/etc/zabbix/zabbix_server.conf`;
- `/etc/zabbix/web/zabbix.conf.php`.

Note:

Usuários individuais `zabbix_srv` e `zabbix_web` são definidos para o Server e o Frontend respectivamente.

2.2 Localização de arquivos

- Arquivos de configuração estão localizados em **`/etc/zabbix`**.
- Arquivos de log para Zabbix Server, Proxy e Agent estão localizados em **`/var/log/zabbix`**.

- O Zabbix Frontend está localizado em **/usr/share/zabbix**.
- O diretório Home para o usuário **zabbix** é **/var/lib/zabbix**.

2.3 Alterações às configurações do Zabbix

- O timezone do Frontend está configurado para Europe/Riga (isto pode ser modificado em **/etc/php-fpm.d/zabbix.conf**);

3 Acesso ao Frontend Por padrão, o acesso ao frontend é permitido de qualquer local.

O frontend pode ser acesso em `http://<host>`.

Isto pode ser customizado em **/etc/nginx/conf.d/zabbix.conf**. O Nginx deve ser reiniciado após alterações neste arquivo. Para tal, acesse via SSH como usuário **root** e execute:

```
systemctl restart nginx
```

4 Firewall Por padrão, apenas as portas listadas nas **mudanças de configuração** acima estão abertas. Para abrir portas adicionais, modifique o arquivo `/etc/sysconfig/iptables` e recarregue as regras de firewall:

```
systemctl reload iptables
```

5 Atualizando Os pacotes da aplicação do Zabbix podem ser atualizados. Para fazê-lo, execute:

```
dnf update zabbix*
```

6 Serviços de Sistema Serviços compatíveis com Systemd estão disponíveis:

```
systemctl list-units zabbix*
```

7 Notas específicas de formatação 7.1 VMware

As imagens no formato vmdk são utilizáveis diretamente nos produtos VMware Player, Server e Workstation. Para uso em ESX, ESXi e vSphere elas devem ser convertidas usando [VMware converter](#).

7.2 HDD/flash image (raw)

```
dd if=./zabbix_appliance_5.2.0.raw of=/dev/sdc bs=4k conv=fdatasync
```

Substitua `/dev/sdc` por seu dispositivo de disco Flash/HDD.

7. Configuração

Por favor, use a barra lateral para acessar o conteúdo na seção de Configuração.

1 Configurando um modelo

Visão geral

A configuração de um modelo requer que você primeiro crie um modelo definindo seus parâmetros gerais e então você adiciona entidades (itens, gatilhos, gráficos, etc.)

Criando um modelo

Para criar um modelo, faça o seguinte:

- Vá para Configuração → Modelos
- Clique em Criar modelo
- Editar atributos do modelo

A guia **Modelos** contém atributos gerais de modelo.

Template
Tags 1
Macros 9
Value mapping

* Template name
Elasticsearch

Visible name
Elasticsearch

Templates
type here to search
Select

* Groups
Templates/Applications ✕
type here to search
Select

Description

Add Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Atributos do modelo:

Parâmetro	Descrição
Nome do modelo	Nome do modelo exclusivo. Alfanuméricos, espaços, pontos, traços e sublinhados são permitidos. No entanto, espaços à esquerda e à direita não são permitidos.
Nome visível	Se você definir este nome, ele será o visível em listas, mapas, etc.
Modelos	Vincule um ou mais modelos "aninhados" a este modelo. Todas as entidades (itens, acionadores, gráficos, etc.) serão herdadas dos modelos vinculados. Para vincular um novo modelo, comece a digitar o nome do modelo no campo Vincular novos modelos. Uma lista de modelos correspondentes aparecerá; role para baixo para selecionar. Alternativamente, você pode clicar em Selecionar ao lado do campo e selecionar modelos da lista em uma janela pop-up. Os modelos selecionados no campo Vincular novos modelos serão vinculados ao modelo quando o formulário de configuração do modelo for salvo ou atualizado. Para desvincular um modelo, use uma das duas opções no bloco Modelos vinculados : Desvincular - desvincular o modelo, mas preservar seus itens, acionadores e gráficos Desvincular e limpar - desvincular o modelo e remover todos os seus itens, acionadores e gráficos
Grupos	Grupos de hosts/modelos aos quais o modelo pertence.
Descrição	Digite a descrição do modelo.

A guia **Tags** permite definir o nível do modelo **tags**. Todos os problemas de hosts ligados a este modelo será marcado com os valores inseridos aqui.

Template
Tags 1
Macros 9
Value mapping

Name
Value

App
MySQL

tag
value

Add

Macros de usuário, macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, As macros {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} e {HOST.ID} são suportado em tags.

A guia **Macros** permite definir **user macros** como pares nome-valor. Observação que os valores de macro podem ser mantidos como texto simples, texto secreto ou Vault segredo. A adição de uma descrição também é suportada.

Template Tags 1 Macros 9 Value mapping

Template macros Inherited and template macros

Macro	Value		Description
{TEMPLATE_THRESHOLD1}	10M	T	description
{TEMPLATE_THRESHOLD2}	20M	T	description
{TEMPLATE_THRESHOLD3}	30M	T	description
{TEMPLATE_THRESHOLD4}	40M	T	description
{TEMPLATE_THRESHOLD5}	50M	T	description

Você também pode ver aqui macros de modelos vinculados e macros globais se você seleciona a opção Macros herdados e de modelo. É aí que todos macros de usuário definidas para o modelo são exibidas com o valor que resolver, bem como a sua origem.

Template Tags 1 Macros 9 Value mapping

Template macros Inherited and template macros

Macro	Effective value	
{\$AGENT.TIMEOUT}	3m	T
Timeout after which agent is considered unavailable. Works only for agents reachable from Zabbix server/proxy (passive mode).		
{\$CPU.UTIL.CRIT}	90	T
description		
{\$IF.ERRORS.WARN}	2	T
description		
{\$IF.CONTROL}	1	T

Por conveniência, links para os respectivos modelos e macro global configuração são fornecidos. Também é possível editar um aninhado modelo/macro global no nível do modelo, criando efetivamente uma cópia da macro no modelo.

A guia **Mapeamento de valor** permite configurar representação dos dados do item em **valor mapeamentos**.

Botões:

Add

Update

Clone

Full clone

Adicione o modelo. O modelo adicionado deve aparecer na lista.

Atualize as propriedades de um modelo existente.

Crie outro template baseado nas propriedades do template atual, incluindo as entidades (itens, triggers, etc) herdado de modelos vinculados.

Crie outro template baseado nas propriedades do template atual, incluindo as entidades (itens, triggers, etc) ambos herdados de modelos vinculados e anexados diretamente ao modelo atual.

Delete	Excluir o modelo; entidades do modelo (itens, gatilhos, etc) permanecem com os hosts vinculados.
Delete and clear	Excluir o modelo e todas as suas entidades dos hosts vinculados.
Cancel	Cancelar a edição das propriedades do template.

Com um template criado, é hora de adicionar algumas entidades a ele.

::: não importante Os itens devem ser adicionados a um modelo primeiro. Triggers e gráficos não podem ser adicionados sem o correspondente item. :::

Adicionando itens, gatilhos, gráficos

Para adicionar itens ao modelo, faça o seguinte:

- Vá para Configuração → Hosts (ou Modelos)
- Clique em Itens na linha do host/modelo necessário
- Marque as caixas de seleção dos itens que deseja adicionar ao modelo
- Clique em Copiar abaixo da lista de itens
- Selecione o modelo (ou grupo de modelos) que os itens devem ser copiado e clique em Copiar

Todos os itens selecionados devem ser copiados para o modelo.

A adição de gatilhos e gráficos é feita de maneira semelhante (da lista de gatilhos e gráficos, respectivamente), novamente, tendo em mente que eles só pode ser adicionado se os itens necessários forem adicionados primeiro.

Adicionando painéis

Para adicionar painéis a um modelo em Configuração → Modelos, faça o Segue:

- Clique em Painéis na linha do modelo
- Configure um painel seguindo as diretrizes de [configurando painéis](#)

Attention:

Os widgets que podem ser incluídos em um modelo painel são: Relógio, Gráfico (clássico), Protótipo de gráfico, Valor do item, Texto simples, URL.

Note:

Para obter detalhes sobre como acessar painéis de host que são criado a partir de painéis de modelo, consulte o [host painel](#) seção.

Configurando regras de descoberta de baixo nível

Consulte a [descoberta de baixo nível](#) seção do manual.

Adicionando cenários da web

Para adicionar cenários da Web a um modelo em Configuração → Modelos, faça a seguir:

- Clique em Web na linha do modelo
- Configure um cenário da web seguindo o método usual de [configurar cenários da web](#)

2 Vinculando/desvinculando

Visão geral

A vinculação é um processo pelo qual os modelos são aplicados aos hosts, enquanto desvincular remove a associação com o modelo de um host.

::: não importante Os modelos são vinculados diretamente a hosts individuais e não para hospedar grupos. Simplesmente adicionar um modelo a um grupo de hosts não o vincule. Os grupos de hosts são usados apenas para agrupamento lógico de hosts e modelos.

:::

Vinculando um modelo

Para vincular um modelo ao host, faça o seguinte:

- Vá para Configuração → Hosts
- Clique no host desejado
- Comece a digitar o nome do modelo no campo Modelos. Uma lista de os modelos correspondentes aparecerão; role para baixo para selecionar.
- Alternativamente, você pode clicar em Selecionar ao lado do campo e selecione um ou vários modelos da lista em uma janela pop-up
- Clique em Adicionar/Atualizar no formulário de atributos do host

O host agora terá todas as entidades (itens, gatilhos, gráficos, etc.) do modelo.

::: não importante Vincular vários modelos ao mesmo host falhará se nesses modelos houver itens com a mesma chave de item. E, como gatilhos e gráficos usam itens, eles não podem ser vinculados a um único host de vários modelos, se estiver usando itens idênticos chaves. :::

Quando entidades (itens, gatilhos, gráficos etc.) modelo:

- entidades idênticas previamente existentes no host são atualizadas conforme entidades do modelo
- entidades do modelo são adicionadas
- quaisquer entidades diretamente vinculadas que, antes da vinculação do modelo, existia apenas no host permanecem intocados

Nas listas, todas as entidades do modelo agora são prefixadas pelo nome do modelo, indicando que eles pertencem ao modelo específico. O próprio nome do modelo (em texto cinza) é um link que permite acessar o lista dessas entidades no nível do modelo.

Se alguma entidade (item, gatilho, gráfico etc.) nome do modelo, significa que ele existia no host antes e não foi adicionado pelo modelo.

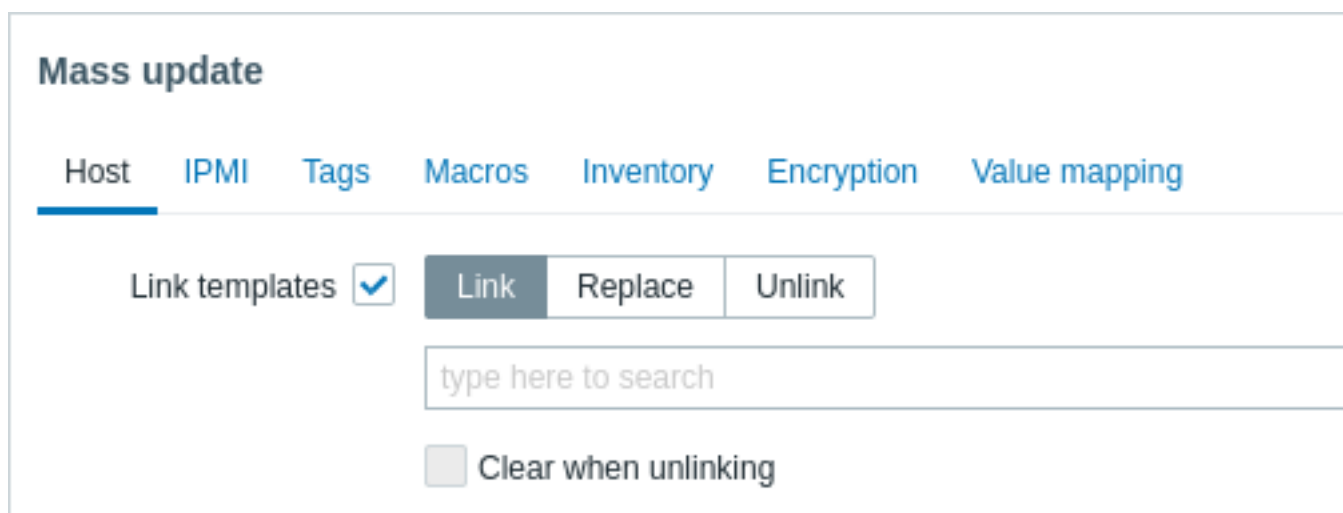
Critérios de exclusividade da entidade

Ao adicionar entidades (itens, gatilhos, gráficos etc.) de um modelo, é importante saber quais dessas entidades já existem no host e precisam ser atualizados e quais entidades diferem. Os critérios de exclusividade para decidir sobre a semelhança/diferença são:

- para itens - a chave do item
- para gatilhos - nome e expressão do gatilho
- para gráficos personalizados - nome do gráfico e seus itens

Vinculando modelos a vários hosts

Para atualizar a vinculação de modelos de muitos hosts, em Configuração → Hosts selecione alguns hosts marcando suas caixas de seleção e clique em **Mass update** abaixo da lista e depois selecione Modelos de links:



Mass update

Host IPMI Tags Macros Inventory Encryption Value mapping

Link templates ☒ Link Replace Unlink

type here to search

☐ Clear when unlinking

Para vincular modelos adicionais, comece a digitar o nome do modelo no campo de preenchimento automático até que uma lista suspensa apareça oferecendo a correspondência modelos. Basta rolar para baixo para selecionar o modelo para vincular.

A opção Substituir permitirá vincular um novo modelo ao desvincular qualquer modelo que tenha sido vinculado aos hosts antes. A opção Desvincular permitirá especificar quais modelos devem ser desvinculados. O Limpar quando unlinking permitirá não apenas desvincular qualquer link anteriormente templates, mas também remover todos os elementos herdados deles (itens, gatilhos, etc.).

O Zabbix oferece um conjunto considerável de templates predefinidos. Vocês pode usá-los como referência, mas tome cuidado ao usá-los inalterados em produção, pois podem conter muitos itens e pesquisar dados também muitas vezes. Se você sentir vontade de usá-los, ajuste-os para se adequarem a você precisa.

Editando entidades vinculadas

Se você tentar editar um item ou acionador vinculado ao modelo, você pode perceber que muitas opções de teclas estão desabilitadas para edição. Este faz sentido, pois a ideia de templates é que as coisas são editadas em maneira de um toque no nível do modelo. No entanto, você ainda pode, por exemplo, habilitar/desabilitar um item no host individual e definir o intervalo de atualização, comprimento do histórico e alguns outros parâmetros.

Se você deseja editar a entidade completamente, você deve editá-la no nível de modelo (atalho de nível de modelo é exibido no nome do formulário), tendo em mente que essas alterações afetarão todos os hosts que possuem esse modelo vinculado a eles.

Desvinculando um modelo

Para desvincular um modelo de um host, faça o seguinte:

- Vá para Configuração → Hosts
- Clique no host desejado e encontre o campo Modelos
- Clique em Desvincular ou Desvincular e limpar ao lado do modelo para desvincular
- Clique em Atualizar no formulário de atributos do host

Escolher a opção Desvincular simplesmente removerá a associação com o template, deixando todas as suas entidades (itens, triggers, gráficos etc.) com o anfitrião.

Escolher a opção Desvincular e limpar removerá a associação com o template e todas as suas entidades (itens, triggers, gráficos etc.).

3 Aninhamento

Visão geral

O aninhamento é uma maneira de um modelo abranger um ou mais outros modelos.

Como faz sentido separar entidades de templates individuais para vários serviços, aplicativos, etc., você pode acabar com alguns templates, todos os quais podem precisar ser vinculados a alguns hosts. Para simplificar a imagem, é possível vincular alguns modelos juntos, em um modelo "aninhado".

A vantagem do aninhamento é que você precisa vincular apenas um template para o host e o host herdará todas as entidades dos modelos vinculados automaticamente.

Configurando um modelo aninhado

Se você deseja vincular alguns modelos, para começar, você pode fazer um modelo existente ou um novo, então:

- Abra o formulário de propriedades do modelo
- Encontre o campo Modelos
- Clique em Selecionar para selecionar modelos na janela pop-up
- Clique em Adicionar para listar os modelos selecionados
- Clique em Adicionar/Atualizar no formulário de propriedades do modelo

Assim, todas as entidades do próprio template, bem como todas as entidades vinculadas entidades dos modelos (como itens, gatilhos, gráficos etc.) na configuração do modelo, exceto para o modelo vinculado painéis, que, no entanto, serão herdados pelos hosts.

Para desvincular qualquer um dos modelos vinculados, no mesmo formulário, use o botão Desvincular ou botões Desvincular e limpar e clique em Atualizar.

Escolher a opção Desvincular simplesmente removerá a associação com o outro modelo, sem remover todas as suas entidades (itens, gatilhos, gráficos etc.).

Escolher a opção Desvincular e limpar removerá a associação com o outro template e todas as suas entidades (itens, triggers, gráficos etc.).

4 Atualização em massa

Visão geral

Às vezes você pode querer mudar algum atributo para um número de modelos de uma só vez. Em vez de abrir cada modelo individual para edição, você pode usar a função de atualização em massa para isso.

Usando atualização em massa

Para atualizar alguns modelos em massa, faça o seguinte:

- Marque as caixas de seleção antes dos modelos que deseja atualizar no **modelo lista**
- Clique em Atualização em massa abaixo da lista
- Navegue até a guia com os atributos necessários (Modelo, Tags, Macros ou Mapeamento de valores)
- Marque as caixas de seleção de qualquer atributo para atualizar e insira um novo valor para eles

Mass update

Template

Tags

Macros

Value mapping

Link templates

☒

Link

Replace

Unlink

Select

☐ Clear when unlinking

Host groups

☒

Add

Replace

Remove

Select

Description

☐

Original

Update

Cancel

As seguintes opções estão disponíveis ao selecionar o respectivo botão para atualização de vinculação **modelo**:

- Link - especifique quais modelos adicionais vincular
- Substituir - especifique quais modelos vincular ao desvincular qualquer modelo que estava vinculado aos modelos anteriores
- Desvincular - especifique quais modelos serão desvinculados

Para especificar os modelos para vincular/desvincular comece a digitar o nome do modelo no campo de preenchimento automático até que uma lista suspensa apareça oferecendo o modelos correspondentes. Basta rolar para baixo para selecionar o modelo necessário.

A opção Limpar ao desvincular permitirá não apenas desvincular qualquer templates previamente vinculados, mas também remover todos os elementos herdados de (itens, gatilhos, etc.).

As seguintes opções estão disponíveis ao selecionar o respectivo botão para atualização do **grupo de hosts**:

- Add - permite especificar grupos de hosts adicionais a partir do ou insira grupos de hosts completamente novos para os modelos
- Substituir - removerá o modelo de qualquer grupo de hosts existente e substitua-os pelos especificados neste campo (existentes ou novos grupos de hosts)
- Remover - removerá grupos de hosts específicos dos modelos

Esses campos são de preenchimento automático - começar a digitá-los oferece uma lista suspensa de grupos de hosts correspondentes. Se o grupo de hosts for novo, ele também aparece na lista suspensa e é indicado por (novo) após a string. Basta rolar para baixo para selecionar.

Mass update

Template

Tags

Macros

Value mapping

Tags ☒

Add

Replace

Remove

Name

Value

tag

value

Add

Macros de usuário, macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, As macros {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} e {HOST.ID} são suportado em tags. Observe que as tags com o mesmo nome, mas diferentes valores não são considerados 'duplicados' e podem ser adicionados ao mesmo modelo.

Mass update

Template

Tags

Macros

Value mapping

Macros ☒

Add

Update

Remove

Remove all

Macro

Value

Description

{SMACRO}

value

T v

description

Add

☐ Update existing

As seguintes opções estão disponíveis ao selecionar o respectivo botão para atualização de macros:

- Add - permite especificar macros de usuário adicionais para os modelos. Se a caixa de seleção Atualizar existente estiver marcada, valor, tipo e descrição do nome da macro especificada será atualizada. Se desmarcada, se uma macro com esse nome já existir no template(s), ele não será atualizado.
- Update - substituirá valores, tipos e descrições de macros especificado nesta lista. Se a caixa de seleção Adicionar ausente estiver marcada, a macro que não existia anteriormente em um modelo será adicionado como novo macro. Se desmarcado, apenas macros que já existem em um modelo será atualizado.
- Remove - removerá macros especificadas dos modelos. Se Exceto selecionada está marcada, todas as macros exceto as especificadas na lista será removido. Se desmarcado, apenas macros especificadas na lista será removido.
- Remove tudo - removerá todas as macros de usuário dos modelos. Se eu confirme para remover todas as macros* caixa de seleção não está marcada, um novo pop-up janela será aberta pedindo para confirmar a remoção de todas as macros.

Mass update

Template Tags Macros Value mapping

Value mapping ☒

Add

Update

Rename

Remove

Remove all

Name

Value

Add Add from

☐ Update existing

Botões com as seguintes opções estão disponíveis para atualização do mapa de valores:

- Add - adiciona mapas de valor aos modelos. Se você marcar Atualizar existindo, todas as propriedades do mapa de valores com este nome serão Atualizada. Caso contrário, se já existir um mapa de valores com esse nome, ele não será atualizado.
- Update - atualiza os mapas de valor existentes. Se você marcar Adicionar ausente, um mapa de valor que não existia anteriormente em um modelo será adicionado como um novo mapa de valor. Caso contrário, apenas os mapas de valores que já existem em um modelo será atualizado.
- Renomear - dá um novo nome a um mapa de valores existente
- Remover - remove os mapas de valores especificados dos modelos. Se você marca Exceto selecionado, todos os mapas de valores serão removidos **exceto** os especificados.
- Remover tudo - remove todos os mapas de valores dos modelos. Se o eu confirme para remover todos os mapas de valores não está marcado, um novo janela pop-up será aberta pedindo para confirmar a remoção.

Quando terminar todas as alterações necessárias, clique em Atualizar. Os atributos será atualizado de acordo para todos os modelos selecionados.

1 Hosts e grupos de host

O que é um "host"?

No Zabbix, um "host" refere-se a qualquer dispositivo físico ou virtual, aplicação, serviço, ou qualquer outra coleção logicamente relacionada de parâmetros monitorados.

Criar hosts é uma das primeiras tarefas de monitoramento no Zabbix. Por exemplo, se você deseja monitorar alguns parâmetros em um servidor "x", você deve criar um host chamado, digamos, "Server X" e em seguida, adicionar itens de monitoramento para esse host.

Hosts são organizados em grupos de hosts.

Vá para [creating and configuring a host](#).

1 Configurando um host

Visão Geral

Para configurar um host no Zabbix frontend, siga os passos abaixo:

- Acesse: Configuração→ Hosts ou Monitoramento→ Hosts
- Clique em Criar host à direita (ou no nome do host para editar um host existente)
- Insira os parâmetros do host no formulário.

Você também pode usar os botões Clonar and Clone completo no formulário de um host existente para criar um novo host. Ao clicar em Clonar, serão mantidos todos os parâmetros do host e a vinculação do template (mantendo todas as entidades desses templates. Clone completo também manterá as entidades diretamente anexadas (tags, itens, triggers, gráficos, regras de descoberta de baixo nível e cenários da web).

Note: Quando um host é clonado, ele manterá todas as entidades do template como estão originalmente no template. Quaisquer alterações nessas entidades feitas no nível do host existente (como intervalo de item alterado, expressão regular modificada ou

protótipos adicionados à regra de descoberta de baixo nível) não serão clonadas para o novo host; em vez disso, eles serão como no template.

Configuração

A aba **Host** contém os atributos gerais do host:

Host

Host

IPMI

Tags

Macros 2

Inventory ●

Encryption

Value mapping 1

* Host name

Zabbix server

Visible name

Zabbix server

Templates

Name

Action

Linux OS agent

Unlink

Unlink and clear

App Zabbix Server

Unlink

Unlink and clear

type here to search

* Groups

Zabbix servers X

type here to search

Interfaces

Type

IP address

DNS name

Agent

127.0.0.1

SNMP

127.0.0.1

Add

Description

Monitored by proxy

(no proxy)

Enabled

☒

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome do host	Informe um nome de host único. Caracteres alfanuméricos, espaços, pontos, traços e sublinhados são permitidos. No entanto, espaços antes e depois do nome não são permitidos. Note: Com Zabbix Agent sendo executado no host que você está configurando, o arquivo de configuração do host deve ter o parâmetro Hostname configurado com o mesmo valor que o nome do host informado aqui. O nome no parâmetro é necessário no processamento das verificações ativas .

165

Parâmetro	Descrição
Nome Visível	Informe um nome visível único para o host. Se você configurar este nome, ele será o nome visível nas listas, mapas, etc., em vez do nome do host do parâmetro anterior. Este atributo tem suporte a UTF-8.
Templates	<p>Associe templates ao host. Todas as entidades (itens, triggers, gráficos, etc.) serão herdadas do template.</p> <p>Para associar um novo template, comece digitando o nome do template no campo Vincular novos templates. Uma lista de templates equivalentes será mostrada; role a lista para selecionar. Alternativamente, você pode clicar em Selecionar próximo ao campo e selecionar templates da lista na janela apresentada. Os templates que são selecionados no campo Vincular novos templates serão associados ao host quando o formulário de configuração do host for salvo ou atualizado.</p> <p>Para desvincular um template, use uma das duas opções no bloco Associado aos templates:</p> <p>Desassociar - desvincula o template, mas preserva seus itens, triggers e gráficos;</p> <p>Desassociar e limpar - desvincula o template e remove todos os seus itens, triggers e gráficos.</p> <p>Os nomes de template na lista são clicáveis levando até o formulário de configuração dos templates.</p>
Grupos	<p>Selecione os grupos de host aos quais o host pertence. Um host deve pertencer a um grupo no mínimo. Um novo grupo pode ser criado e vinculado ao grupo do host pela adição de um nome de grupo não existente.</p>
Interfaces	<p>Vários tipos de interface são suportados para um host: Agent, SNMP, JMX e IPMI.</p> <p>Nenhuma interface é definida por padrão. Para adicionar uma nova interface, clique em Adicionar no bloco de Interfaces, selecione o tipo de interface e informe IP/DNS, Conectado a e Porta.</p> <p>Nota: Interfaces que são usadas em algum item não podem ser removidas e o link Remover aparecerá cinza para elas.</p> <p>Consulte configuração de monitoramento SNMP para detalhes adicionais na configuração de uma interface SNMP (v1, v2 e v3).</p>
Endereço IP	Endereço IP do host (opcional).
Nome DNS	Nome DNS do host (opcional).
Conectar a	<p>Clicando na opção desejada você informa ao Zabbix Server o que utilizar para receber os dados dos Agents:</p> <p>IP - Conecte ao endereço de IP (recomendado)</p> <p>DNS - Conecte ao nome DNS</p>
Porta	Número da porta TCP/UDP. Os valores padrão são: 10050 para o Zabbix Agent, 161 para SNMP Agent, 12345 para JMX e 623 para IPMI.
Padrão	Selecione esta opção para marcar a interface como padrão.
Descrição	Adicione uma descrição para o host.
Monitorado por proxy	<p>O host pode ser monitorado ou pelo Zabbix Server ou por um dos Zabbix Proxies:</p> <p>(sem proxy) - o host é monitorado pelo Zabbix Server</p> <p>Nome do proxy - o host é monitorado pelo Zabbix Proxy "nome do proxy"</p>
Habilitado	Marque esta caixa para tornar o host ativo, pronto para ser monitorado. Se desmarcado, o host não está ativo, portanto não monitorado.

A aba **IPMI** contém os atributos de gerenciamento do IPMI.

Parâmetro	Descrição
Algoritmo de autenticação	Selecione o algoritmo de autenticação.
Nível de acesso	Selecione o nível de acesso.
Usuário	Nome de usuário para autenticação. Macros de usuário podem ser usadas.
Senha	Senha para autenticação. Macros de usuário podem ser usadas.

A aba **Etiquetas** permite definir **etiquetas** para o nível de host. Todos os problemas deste host serão etiquetados com os valores informados aqui.

Host
IPMI
Tags 1
Macros 2
Inventory ●
Encryption
Value mapping 1

Name

Value

Service

JIRA

Add

Macros de usuário, macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} e {HOST.ID} são suportadas nas etiquetas.

A aba **Macros** permite definir, no nível de host, **macros de usuário** como um par nome-valor. Note que os valores das macros podem ser mantidos como texto plano, texto secreto ou Vault secreto. A adição de uma descrição também é suportada.

Host
IPMI
Tags 1
Macros 2
Inventory ●
Encryption
Value mapping 1

Host macros

Inherited and host macros

Macro

Value

{ \$HOST_MACRO }

1

T ▾

{ \$SNMP_COMMUNITY }

public

T ▾

Add

Você também observar aqui macros de template e macros de usuário globais se você selecionar a opção Macros herdadas e do host. É onde todas as macros de usuário definidas para o host são mostradas com o valor que receberam, assim como sua origem.

Host
IPMI
Tags 1
Macros 2
Inventory ●
Encryption
Value mapping 1

Host macros

Inherited and host macros

Macro

Effective value

Templa

{ \$AGENT.TIMEOUT }

3m

T ▾

Change ← Templa

Timeout after which agent is considered unavailable. Works only for agents reachable from Zabbix server/proxy (passive mode).

{ \$CPU_UTIL_CRIT }

90

T ▾

Change ← Templa

description

{ \$HOST_MACRO }

1

T ▾

Remove

Para conveniência, links para os respectivos templates e configuração das macros globais são fornecidos. Também é possível editar um template/macro global no nível do host, efetivamente criando uma cópia da macro no host.

A aba **Inventário** permite adicionar manualmente informações de **inventário** para o host. Você também pode selecionar a opção de habilitar o preenchimento Automático do inventário, ou desabilitar o preenchimento de inventário para este host.

[Host](#) [IPMI](#) [Tags 1](#) [Macros 2](#) [Inventory ●](#) [Encryption](#) [Value mapping 1](#)

Disabled Manual Automatic

Type Zabbix server

Type (Full details)

Se o inventário está habilitado (manual ou automático), um sinal verde é mostrado junto ao nome da aba.

Criptografia

A aba **Criptografia** permite que você estabeleça conexões **criptografadas** com o host.

Parâmetro	Descrição
Conexões com o host	Como o Zabbix Server ou Proxy se conecta ao Zabbix Agent em um host: sem criptografia (padrão), usando PSK (chave pré-compartilhada) ou certificado.
Conexões do host	Selecione quais tipos de conexões são permitidas a partir do host (p.e. do Zabbix Agent e Zabbix Sender). Vários tipos de conexões podem ser selecionados simultaneamente (útil para testes e alteração para outro tipo de conexão). O padrão é "Sem criptografia".
Emissor	Emissor de certificado permitido. Um certificado é primeiramente validado com uma CA (autoridade de certificado). Se for válido, assinado pela CA, então o campo Emissor pode ser usado para futuras restrições de CAs permitidos. O objetivo de uso deste campo é para quando sua instalação do Zabbix utilizar certificados de múltiplos CAs. Se este campo estiver vazio então nenhum CA é aceito.
Segurado	Segurado de certificado permitido. Um certificado é primeiramente validado com uma CA. Se for válido, assinado pela CA, então o campo Segurado pode ser usado para permitir apenas um valor (formato texto) de segurado. Se este campo estiver vazio então qualquer certificado válido assinado pela CA configurada é aceito.
Identidade PSK	Texto da identidade de chave pré-compartilhada. Não coloque informações sensíveis na identidade PSK, ela é transmitida de forma não criptografada pela rede para informar um receptor qual PSK utilizar.
PSK	Chave pré-compartilhada (texto hexadecimal). Largura máxima: 512 dígitos hexadecimais (256-byte PSK) se o Zabbix usa biblioteca GnuTLS ou OpenSSL, 64 dígitos hexadecimais (32-byte PSK) se o Zabbix usa biblioteca mbed TLS (PolarSSL). Exemplo: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952

Mapeamento de valor

A aba de **Mapeamento de valor** permite configurar representações humanizadas dos dados de itens com **mapeamento de valor**.

Criando um grupo de host

Attention:

Apenas usuários Super Admin podem criar grupos de hosts.

Para criar um grupo de host no frontend do Zabbix, siga os passos abaixo:

- Acesse: Configuração → Grupos de hosts
- Clique em Criar Grupo no canto superior direito da tela
- Insira os parâmetros do grupo no formulário

≡ Host groups

* Group name

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome do grupo	<p>Digite um nome único para o grupo de host.</p> <p>Para criar um grupo de host aninhado, use o separador '/' barra inclinada, por exemplo Europe/Latvia/Riga/Zabbix servers. Você pode criar este grupo mesmo que nenhum dos três grupos de host pai (Europe/Latvia/Riga) exista. Nesse caso, a criação desses grupos de host pai fica a critério do usuário; eles não serão criados automaticamente.</p> <p>As barras inclinadas no início e no final, várias barras inclinadas seguidas não são permitidas. O escape de '/' não é suportado..</p> <p>A representação aninhada de grupos de host é suportada desde o Zabbix 3.2.0.</p>
Aplicar permissões e filtros de tags a todos os subgrupos	<p>Esta caixa de seleção está disponível apenas para usuários Super Admin e apenas ao editar um grupo de host existente.</p> <p>Marque esta caixa de seleção e clique em Atualizar para aplicar o mesmo nível de permissões/filtros de tags a todos os grupos de host aninhados. Para grupos de usuários que possam ter tido diferentes permissões permissions atribuídas a grupos de host aninhados, o nível de permissão do grupo de host pai será aplicado aos grupos aninhados.</p> <p>Esta é uma opção única que não é salva no banco de dados.</p> <p>Esta opção é suportada desde o Zabbix 3.4.0..</p>

Permissões para grupos de host aninhados

- Ao criar um grupo de host filho para um grupo de host pai existente, **user group** as permissões do grupo de usuários para o filho são herdadas do pai (por exemplo, ao criar Riga/Zabbix servers se Riga já existir)
- Ao criar um grupo de host pai para um grupo de host filho existente, nenhuma permissão é definida para o pai (por exemplo, ao criar Riga se Riga/Zabbix servers já existir)

2 Inventário

Visão geral

Você pode manter o inventário de dispositivos em rede no Zabbix.

Há um menu especial Inventário no frontend do Zabbix. Entretanto, você não visualizará nenhuma informação inicialmente e não é onde você insere os dados. A criação de um inventário de dados é feita manualmente quando configurado um host ou automaticamente através do uso de alguma opção de preenchimento automático.

Construindo o inventário

Modo manual

Ao **configuring a host**, na aba Inventário, você pode inserir detalhes como o tipo de dispositivo, número de série, localização, pessoa responsável, etc - dados que irão preencher as informações do inventário.

Se uma URL for incluída nas informações de inventário do host e iniciar com 'http' ou 'https', resultará em um link clicável na sessão Inventário.

Modo automático

O Inventário do host também pode ser preenchido automaticamente. Para que isso funcione, na configuração do host, na aba Inventário, o modo de inventário deve ser selecionado como Automático.

Desta forma, você pode **configure host items** para popular qualquer campo do inventário do host com seu valor, indicando o campo de destino com seu respectivo atributo (chamado Item que preencherá o campo de inventário do host) na configuração do item.

Itens que são úteis na automatização do inventário:

- system.hw.chassis[full|type|vendor|model|serial] - o padrão é [full], permissões privilegiadas (root) são necessárias
- system.hw.cpu[all|cpunum,full|maxfreq|vendor|model|curfreq] - o padrão é [all,full]
- system.hw.devices[pci|usb] - o padrão é [pci]
- system.hw.macaddr[interface,short|full] - o padrão é [all,full], interface é uma expressão regular (regexp)
- system.sw.arch
- system.sw.os[name|short|full] - o padrão é [name]
- system.sw.packages[package,manager,short|full] - o padrão é [all,all,full], package é uma expressão regular (regexp)

Seleção do modo de inventário

O modo de inventário pode ser selecionado na configuração do host.

O modo de inventário por padrão, para novos hosts, é selecionado com base no Modo padrão do inventário em novos hosts definido em Administração → Geral → **Other**.

Para hosts adicionados a partir de regras de descoberta de rede ou ações de autorregistro, é possível definir uma Operação de modo de inventário selecionando modo manual ou automático. Esta operação sobrescreve as configurações de Modo padrão de inventário de host.

Visão geral do inventário

Os detalhes de todos os dados presentes no inventário estão disponíveis no menu Inventário.

Em Inventário → Visão Geral você visualizar a quantidade de hosts filtrando por vários campos do inventário.

Em Inventário → Hosts você pode visualizar todos os hosts que contém informações de inventário. Ao clicar no nome do host, serão exibidos os detalhes do inventário do host em questão.

≡ Host inventory

OverviewDetails

Host nameZabbix server

Agent interfaces

IP address	DNS name	Connect to	Port
127.0.0.1		IPDNS	10050

SNMP interfaces

127.0.0.1		IPDNS	161
-----------	--	-------	-----

OS

Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP

Monitoring

WebLatest dataProblemsGraphsDashboards

Configuration

HostItems 148Triggers 67Graphs 28Discovery 4Web 1

Cancel

A aba **Visão geral** apresenta:

Parâmetro	Descrição
Nome do host	Nome do Host. Ao clicar no nome do host é aberto o menu com os scripts definidos para ele. O nome do host é exibido com um ícone laranja caso esteja em modo de manutenção.
Nome visível	Nome visível do host (se definido).

Parâmetro	Descrição
Interfaces do Host (Agent, SNMP, JMX, IPMI)	Este bloco mostra os detalhes das interfaces configuradas para o host.
OS	Sistema Operacional definido no inventário (se definido).
Hardware	Descrição do hardware (se definido).
Software	Descrição do software definido (se definido).
Descrição	Descrição do Host.
Monitoramento	Links para informações de monitoramento do host: Web, Dados recentes, Problemas, Gráficos, Dashboards.
Configuração	Links para configurações do host: Host, Aplicações, Items, Triggers, Gráficos, Descobertas, Web. A quantidade de entidades configuradas de cada tipo é exibida.

A aba **Detalhes** exibe todos os campos que possuem valores (que não estão vazios).

Macros de inventário

Existem macros de inventário de host {INVENTORY.*} disponíveis para utilização em notificações, por exemplo:

"O Servidor em {INVENTORY.LOCATION1} apresenta um problema, a pessoa responsável é {INVENTORY.CONTACT1}, número de telefone {INVENTORY.POC.PRIMARY.PHONE.A1}."

Para mais detalhes, acesse a página [supported macro](#).

3 Atualização em massa

Visão geral

As vezes será necessário modificar algum atributo de vários hosts de uma só vez. Ao invés de abrir cada host individualmente para editá-lo, você pode utilizar o recurso de atualização em massa fazer isto.

Usando a atualização em massa

Para realizar a atualização em massa de alguns hosts, faça o seguinte:

- Selecione os hosts que você deseja atualizar em [host list](#)
- Clique em Atualização em Massa situado abaixo da lista
- Navegue pelas abas de atributos (Host, IPMI, Tags, Macros, Inventário, Criptografia or Mapeamento de valores)
- Selecione o(s) atributo(s) que deseja atualizar e insira o novo valor para este(s)

Mass update

Host

IPMI

Tags

Macros

Inventory

Encryption

Value mapping

Link templates

☒

Link

Replace

Unlink

type here to search

Select

☐ Clear when unlinking

Host groups

☒

Add

Replace

Remove

type here to search

Select

Description

☐ Original

Monitored by proxy

☐ Original

Status

☐ Original

Update

Cancel

As opções abaixo estarão disponíveis ao selecionar o botão correspondente para a atualização de **template** vinculado:

- Associar - especifica quais templates adicionais a serem vinculados
- Substituir - especifica quais templates vincular quando algum template anteriormente associado ao host for desvinculado
- Desassociar - especifica quais templates serão desvinculados

Para especificar quais templates associar/desassociar, comece a digitar o nome do template até que o recurso de autocompletar exiba uma listagem com os templates correspondentes. Basta então selecionar o template desejado.

A opção Limpar ao desassociar, além de desvincular templates associados anteriormente, permite também a remoção de todos os elementos herdados do template (itens, triggers, etc.).

As opções a seguir estão disponíveis ao selecionar o botão para atualização do **Grupo do host**:

As seguintes opções estão disponíveis ao selecionar o respectivo botão para atualização do grupo de hosts:

- Adicionar - permite especificar grupos de host adicionais dentre os já existentes ou inserir um grupo totalmente novo para os hosts
- Substituir - remove do host o(s) grupo(s) existente(s) e o(s) substitui pelo(s) grupo(s) especificado(s) (existentes ou novos)
- Remover - remove o host do(s) grupo(s) especificados

INCOMPLETO

2 Itens

Visão geral

Um item é uma métrica individual.

Os itens são usados para coletar dados. Depois de configurar um host, você precisa adicionar alguns itens para que você possa obter dados reais. Uma maneira de adicionar rapidamente muitos itens é anexar uma das templates predefinidas a um host. No entanto, para um desempenho otimizado do sistema, você pode precisar ajustar as templates para ter os itens e a frequência de monitoramento necessária.

Para especificar qual tipo de dado coletar de um host, utilize o **item key**.

172

Por exemplo, um item com o nome key **system.cpu.load** coletará dados de carga do processador, enquanto um item de com o nome **net.if.in** coletará informações de tráfego de entrada.

Parâmetros adicionais podem ser especificados entre colchetes depois do nome key. Por exemplo, **system.cpu.load[avg5]** retornará a média de carga do processador nos últimos 5 minutos, enquanto **net.if.in[eth0]** mostrará o tráfego de entrada na interface "eth0".

Note:

Consulte seções individuais de **item types** para todos os tipos de itens e itens keys.

Prossiga para **creating and configuring an item**.

Tipos de Item

Os tipos de itens abrangem vários métodos de aquisição de dados do seu sistema. Cada tipo de item vem com seu próprio conjunto de chaves de item suportadas e parâmetros necessários.

Os seguintes tipos de itens são oferecidos atualmente pelo Zabbix:

- **·verificações do agente Zabbix**
- **·verificações do agente SNMP**
- **·armadilhas SNMP**
- **·verificações de IPMI**
- **·Verificações simples -- ·VMware ·monitoramento**
- **·Monitoramento do arquivo de log**
- **·Itens calculados -- ·Agregado ·cálculos**
- **·verificações internas do Zabbix**
- **·verificações SSH**
- **·Telnet checks**
- **·Verificações externas**
- **·Itens do Trapper**
- **·Monitoramento JMX**
- **·verificações ODBC**
- **·Itens dependentes**
- **·verificações HTTP**
- **·verificações do Prometheus**
- **·Itens de script**

Detalhes para todos os tipos de itens estão incluídos nas subpáginas desta seção. Embora os tipos de itens ofereçam muitas opções para coleta de dados, há outras opções através de **userparameters** ou **loadablemodules**.

Algumas verificações são realizadas apenas pelo servidor Zabbix (como sem agente de monitoramento) enquanto outros requerem agente Zabbix ou mesmo Zabbix Java gateway (com monitoramento JMX).

::: nota importante Se um determinado tipo de item requer uma determinada interface (como uma verificação de IPMI precisa de uma interface IPMI no host) que a interface deve existir na definição do host. :::

Múltiplas interfaces podem ser definidas nas opções do host: Zabbix agent, Agente SNMP, JMX e IPMI. Se um item pode usar mais de uma interface, ele irá procurar as interfaces de host disponíveis (na ordem: Agente→SNMP→JMX→IPMI) para o primeiro a ser vinculado.

Todos os itens que retornarem texto (caracter, log, tipos de texto de informação) também podem retornar espaços em branco (quando aplicável), definindo o retorno value para uma string vazia (com suporte desde 2.0).

COMMENT <https://git.zabbix.com/projects/WEB/repos/documentation/browse/en/manual/config/items/itemtypes.md?at=refs%2Fheads%2F>
CONTEXT manual/config/items/itemtypes.md RESOURCE Documentation 6.0•manual/config/items/itemtypes.xliff

1 Agente do Zabbix

Visão geral

These checks use the communication with Zabbix agent for data gathering.

Existem verificações passivas e ativas do agente **passive and active** . Ao configurar um item, você pode escolher o modelo necessário:

- Agente do Zabbix - para verificações passivas
- Agente do Zabbix (ativo) - para verificações ativas

Chaves de Item Suportadas

A tabela fornece detalhes sobre as chaves de item que você pode usar com itens do agente Zabbix agrupados pela família do item.

Consulte também:

- [Items supported by platform](#)
- [Item keys supported by Zabbix agent 2](#)
- [Item keys specific for Windows agent](#)
- [Minimum permission level for Windows agent items](#)

Parâmetros obrigatórios e opcionais

Parâmetros sem colchetes angulares são obrigatórios. Parâmetros marcados com colchetes angulares < > são opcionais.

Uso com utilitários de linha de comando

Observe que ao testar ou usar chaves de item com `zabbix_agentd` ou `zabbix_get` na linha de comando, você também deve considerar a sintaxe do shell.

Por exemplo, se um determinado parâmetro da chave tiver que ser colocado entre aspas duplas, você precisa escapar explicitamente das aspas duplas, caso contrário elas serão cortadas pelo shell como caracteres especiais e não serão passadas para o utilitário Zabbix.

Exemplos:

```
$ zabbix_agentd -t 'vfs.dir.count[/var/log,,,"arquivo,dir",,0]'
```

```
$ zabbix_agentd -t 'vfs.dir.count[/var/log,,,\"arquivo,dir\",,0]'
```

Zabbix metrics

Key	Description	Return value	Parameters	Comments
agent.hostmetadata	Agent host metadata.	String		Returns the value of HostMetadata or HostMetadataItem parameters, or empty string if none are defined. Supported since Zabbix 6.0.
agent.hostname	Agent host name.	String		Returns: As passive check - the name of the first host listed in the Hostname parameter of the agent configuration file; As active check - the name of the current hostname.
agent.ping				

Key				
	Agent availability check.	Nothing - unavailable 1 - available		Use the nodata() trigger function to check for host unavailability.
agent.variant	Variant of Zabbix agent (Zabbix agent or Zabbix agent 2).	Integer		Example of returned value: 1 - Zabbix agent 2 - Zabbix agent 2
agent.version	Version of Zabbix agent.	String		Example of returned value: 6.0.3
zabbix.stats[<ip>,<port>]	Return a set of Zabbix server or proxy internal metrics remotely.	JSON object	ip - IP/DNS/network mask list of servers/proxies to be remotely queried (default is 127.0.0.1) port - port of server/proxy to be remotely queried (default is 10051)	Note that the stats request will only be accepted from the addresses listed in the 'StatsAllowedIP' server/proxy parameter on the target instance. A selected set of internal metrics is returned by this item. For details, see Remote monitoring of Zabbix stats .
zabbix.stats[<ip>,<port>,queue,<from>,<to>]				

Key

	Return number of monitored items in the queue which are delayed on Zabbix server or proxy remotely.	JSON object	ip - IP/DNS/network mask list of servers/proxies to be remotely queried (default is 127.0.0.1) port - port of server/proxy to be remotely queried (default is 10051) queue - constant (to be used as is) from - delayed by at least (default is 6 seconds) to - delayed by at most (default is infinity)	Note that the stats request will only be accepted from the addresses listed in the 'StatsAllowedIP' server/proxy parameter on the target instance.
--	---	-------------	---	---

Resolução de problemas com itens do Agente

- Se for utilizada verificação passiva, o valor de Timeout no arquivo de configuração do Server precisa ser maior do que o Timeout definido no arquivo de configuração do Agent. Caso contrário o item pode não receber nenhum valor devido a requisição do Server ao Agent terminar antes.

Configurações de codificação

Para garantir que os dados adquiridos não sejam corrompidos, você pode especificar a codificação correta para processar a verificação (por exemplo, 'vfs.file.contents') no parâmetro `encoding`. A lista de codificações suportadas (identificadores de página de código) pode ser encontrada na documentação do [libiconv](#) (Projeto GNU) ou na documentação do Microsoft Windows SDK para "Identificadores de Página de Código".

Se nenhuma codificação for especificada no parâmetro `encoding`, as seguintes estratégias de resolução são aplicadas:

- Se a codificação não for especificada (ou for uma string vazia), ela é assumida como UTF-8, os dados são processados "como estão";
- Análise de BOM - aplicável para itens 'vfs.file.contents', 'vfs.file.regexp', 'vfs.file.regmatch'. Uma tentativa é feita para determinar a codificação correta usando a marca de ordem de bytes (BOM) no início do arquivo. Se o BOM não estiver presente, a resolução padrão (consulte acima) é aplicada em vez disso.

Solução de Problemas de Itens do Agente

- Se usado com o agente passivo, o valor de Timeout na configuração do servidor pode precisar ser maior do que o Timeout no arquivo de configuração do agente. Caso contrário, o item pode não receber nenhum valor porque a solicitação do servidor para o agente expirou primeiro.

Dados do Kernel

Chave do Item			
Descrição	Valor de Retorno	Parâmetros	Comentários
kernel.maxfiles			

Chave do Item		
Número máximo de arquivos abertos suportados pelo sistema operacional.	Inteiro	
kernel.maxproc		
Número máximo de processos suportados pelo sistema operacional.	Inteiro	
kernel.openfiles		
Retorna o número de descritores de arquivos atualmente abertos.	Inteiro	Este item é suportado desde o Zabbix 6.0.

Dados de log

Consulte informações adicionais em [log monitoring](#).

Chave do Item			
Descrição	Valor Retornado	Parâmetros	Comentários
log [file,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent_dir>]			

Chave do Item			
Monitoramento de um arquivo de log.	Log	<p>arquivo - caminho completo e nome do arquivo de log</p> <p>regex - expression regular descrevendo o padrão necessário</p> <p>encoding - identificador da página de código identifier</p> <p>maxlines - número máximo de linhas novas que o agente enviará para o servidor ou proxy Zabbix por segundo. Este parâmetro substitui o valor de 'MaxLinesPerSecond' em zabbix_agentd.conf</p> <p>mode (desde a versão 2.0)- valores possíveis:</p> <p>todos (padrão), skip - pular o processamento de dados mais antigos (afeta somente os itens criados recentemente).</p> <p>output (desde a versão 2.2) - um template opcional de formatação de saída. A sequência de escape \0 substituída pela parte correspondente do texto (do primeiro caractere onde a correspondência começa até o caractere onde a correspondência termina), enquanto uma sequência de escape \N (onde N=1...9) é substituída pelo N-ésimo grupo correspondido (ou uma string vazia se o N exceder o número de grupos capturados).</p> <p>maxdelay (desde a versão 3.2) - atraso máximo em segundos. Tipo: float. Valores: 0 - (padrão) nunca ignorar linhas de arquivo de log; > 0.0 - ignorar linhas mais antigas para obter as linhas mais recentes analisadas dentro de "maxdelay" segundos. Leia as notas sobre maxdelay antes de usá-lo!</p> <p>opções (desde a versão 4.4.7) - opções adicionais:</p> <p>mtime-noread - registros não exclusivos, reler apenas se o tamanho do arquivo mudar (ignorar mudança de tempo de modificação). (Este parâmetro está obsoleto desde a versão 5.0.2, pois agora mtime é ignorado.)</p> <p>persistent_dir (desde versões 5.0.18, 5.4.9, somente no zabbix_agentd em Unix systems; não suportado no Agent2) - caminho absoluto do diretório onde armazenar arquivos persistentes. Veja também notas adicionais sobre persistent files.</p>	<p>Este item deve ser configurado como um active check.</p> <p>Se o arquivo estiver ausente ou as permissões não permitirem acesso, o item torna-se não suportado.</p> <p>Se o output estiver vazio - a linha inteira contendo o texto correspondido é retornada. Observe que todos os tipos globais de expressões regulares, exceto 'Result is TRUE' sempre retornam a linha inteira correspondente e o parâmetro output é ignorado.</p> <p>A extração de conteúdo usando o parâmetro output ocorre no agente.</p> <p>Exemplos:</p> <pre>=> log[/var/log/syslog] => log[/var/log/syslog,error] => log[/home/zabbix/logs/logfile,,,100]</pre> <p>Usando o parâmetro output para extrair um número do registro de log:</p> <pre>=> log[/app1/app.log,"task run [0-9.]+ sec, processed ([0-9.]+) records, [0-9.]+ errors",,,1] → corresponderá a um registro de log "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" e enviará apenas '6080' para o servidor. Como um valor numérico está sendo enviado, o "Tipo de informação" para este item pode ser definido como "Numérico (sem sinal)" e o valor pode ser usado em gráficos, triggers, etc.</pre> <p>Usando o parâmetro output para reescrever o registro de log antes de enviar para o servidor:</p> <pre>=> log[/app1/app.log,"([0-9.]+) task run ([0-9.]+) sec, processed ([0-9.]+) records, ([0-9.]+) errors",,,1 RECORDS: \3, ERRORS: \4, DURATION: \2"] → corresponderá a um registro de log "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" e enviará um registro modificado "2015-11-13 10:08:26 RECORDS: 6080, ERRORS: 0, DURATION: 6.08" para o servidor.</pre>

log.count[file,<regex>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent_dir>]

Chave do Item			
Contagem de linhas correspondentes em um arquivo de log monitorado.	Inteiro	<p>arquivo - caminho completo e nome do arquivo de log</p> <p>regexp - expressão regular expression descrevendo o padrão necessário</p> <p>encoding - ### identificador da página de código identifier</p> <p>maxproclines - número máximo de novas linhas por segundo que o agente analisará (não pode exceder 10000). O valor padrão é de 10*MaxLinesPerSecond no zabbix_agentd.conf.</p> <p>mode - Valores possíveis: todos (padrão), skip - pular o processamento de dados mais antigos (afeta somente itens criados recentemente).</p> <p>maxdelay - atraso máximo em segundos. Tipo: float. Valores: 0 - (padrão) nunca ignorar linhas de arquivos de log; > 0.0 - ignorar linhas mais antigas para que as linhas mais recentes sejam analisadas dentro de "maxdelay" segundos. Leia as notas maxdelay antes de usá-lo!</p> <p>opções (desde a versão 4.4.7) - opções adicionais:</p> <p>mtime-noread - registros não únicos, releia apenas se o tamanho do arquivo mudar (ignore a mudança do tempo de modificação). (Este parâmetro está obsoleto desde a versão 5.0.2, porque agora o tempo de modificação é ignorado.)</p> <p>persistent_dir (desde versões 5.0.18, 5.4.9, apenas no zabbix_agentd em Unix systems; não suportado no Agent2) - caminho absoluto do diretório onde armazenar arquivos persistentes. Veja também notas adicionais sobre persistent files.</p>	<p>O item deve ser configurado como uma verificação ativa active check.</p> <p>As linhas correspondentes são contadas nas novas linhas desde a última verificação de log pelo agente e, portanto, dependem do intervalo de atualização do item.</p> <p>Se o arquivo estiver ausente ou as permissões não permitirem o acesso, o item se torna não suportado.</p> <p>Este item não é suportado para Windows Event Log.</p> <p>Suportado desde a versão Zabbix 3.2.0.</p>

logrt[file_regexp,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>,<persistent_dir>]

Chave do Item			
Monitoramento de um arquivo de log que é rotacionado.	Log	<p>file_regexp - caminho absoluto para o arquivo e o nome do arquivo descrito por uma expressão regular expression. Observe que somente o nome do arquivo é uma expressão regular.</p> <p>regexp - expressão regular expression descrevendo o padrão de conteúdo obrigatório.</p> <p>encoding - identificador da página de código identifier</p> <p>maxlines - número máximo de linhas novas por segundo que o agente enviará para servidor ou proxy Zabbix. Esse parâmetro substitui o valor de 'MaxLinesPerSecond' no zabbix_agentd.conf</p> <p>mode (desde a versão 2.0) - valores possíveis: todos (padrão), skip - pular o processamento de dados mais antigos (afeta somente itens criados recentemente).</p> <p>output (desde a versão 2.2) - um template opcional de formatação de saída. A sequência escape \0 é substituída por uma parte correspondente do texto (do primeiro caractere onde a correspondência começa até o caractere onde a correspondência termina) enquanto uma sequência escape \N (where N=1...9) é substituída por um grupo correspondente Nth (ou por uma string vazia caso o N exceder o número de grupos capturados).</p> <p>maxdelay (desde a versão 3.2) - atraso máximo em segundos. Type: float. Valores: 0 - (padrão) nunca ignorar linhas de arquivos de log; > 0.0 - ignorar linhas mais antigas para que as linhas mais recentes sejam analisadas dentro de "maxdelay" segundos. Leia as notas sobre maxdelay antes de usá-lo!</p> <p>options (desde a versão 4.0; opções mtime-reread, mtime-noread desde 4.4.7) - tipo de rotação de arquivo de log e outras opções. Valores possíveis: rotate (padrão), copytruncate - Observe que o copytruncate não pode ser usado junto com o maxdelay. Nesse caso, o maxdelay deve ser 0 ou não especificado; consulte notas sobre copytruncate, mtime-reread - registros não únicos, releia apenas se o tempo ou tamanho mudar (padrão), mtime-noread - registros não únicos, releia apenas se o tamanho do arquivo mudar (ignore a mudança do tempo de modificação).</p> <p>persistent_dir (desde versões 5.0.18, 5.4.9, apenas no zabbix_agentd em Unix systems; não suportado no Agent2) - caminho absoluto do diretório onde armazenar arquivos persistentes. Veja também as notas adicionais sobre persistent</p>	<p>O item deve ser configurado como uma verificação ativa active check. A rotação de log é baseada no tempo da última modificação dos arquivos.</p> <p>Note que logrt foi projetado para funcionar com um arquivo de log atualmente ativo, com vários outros arquivos correspondentes inativos sendo rotacionados. Se, por exemplo, um diretório tiver muitos arquivos de log ativos, um item logrt separado deve ser criado para cada um. Caso contrário, se um item logrt pegar muitos arquivos, pode levar a uma exaustão de memória e a um travamento da monitoração.</p> <p>Se output for deixado em branco - toda a linha contendo o texto correspondente é retornada. Note que todos os tipos de expressões regulares globais, exceto 'O resultado é TRUE', sempre retornam toda a linha correspondente e o parâmetro output é ignorado.</p> <p>A extração de conteúdo usando o parâmetro output ocorre no agente.</p> <p>Exemplos: => logrt["/home/zabbix/logs/^logfile[0-9]{1,3}\$"","",100] → corresponderá a um arquivo como "logfile1" (não corresponderá ".logfile1") => logrt["/home/user/^logfile_.*_[0-9]{1,3}\$","pattern_to_match","UTF-8",100] → coletará dados de arquivos como "logfile_abc_1" ou "logfile__001".</p> <p>Usando o parâmetro <i>output</i> para extrair um número do registro de log: => logrt[/app1/^test.*log\$,"task run [0-9.]+ sec, processed ([0-9.]+) records, [0-9.]+ errors"","",1] → corresponderá a um registro de log "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" e enviará apenas '6080' para o servidor. Como um valor numérico está sendo enviado, o "Tipo de informação" para este item pode ser definido como "Numérico (não assinado)" e o valor pode ser usado em gráficos, triggers etc.</p> <p>*Usando o parâmetro <i>output</i> para reescrever o registro de log antes de enviar para o servidor: => logrt[/app1/^test.*log\$,"([0-9.]+) task run ([0-9.]+) sec, processed ([0-9.]+) records, ([0-9.]+) errors"","",1 RECORDS: \3, ERRORS: \4, DURATION: \2"] → corresponderá a um registro de log "2015-11-13 10:08:26 task run 6.08 sec, processed 6080 records, 0 errors" e enviará um registro modificado "2015-11-13 10:08:26 RECORDS: 6080, ERRORS: 0, DURATION: 6.08" para o</p>

Chave do Item			
logrt.count [file_regexp,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>,<persistent_dir>]	Integer	<p>file_regexp - caminho absoluto para o arquivo e expressão regular expression descrevendo o padrão de nome de arquivo</p> <p>regexp - expressão regular expression descrevendo o padrão de conteúdo requerido</p> <p>encoding - identificador da página de código identifier</p> <p>maxproclines - número máximo de novas linhas por segundo que o agente analisará (não pode exceder 10000). O valor padrão é de 10*MaxLinesPerSecond no zabbix_agentd.conf.</p> <p>mode - valores possíveis: todos (padrão), skip - pular o processamento de dados mais antigos (afeta somente itens criados recentemente)</p> <p>maxdelay - atraso máximo em segundos. Tipo: float. Valores: 0 - (padrão) nunca ignorar linhas de arquivos de log; > 0.0 - ignorar linhas mais antigas para que as linhas mais recentes sejam analisadas dentro de "maxdelay" segundos. Leia as notas sobre maxdelay antes de usá-lo!</p> <p>opções (desde a versão 4.0; opções mtime-reread, mtime-noread desde 4.4.7) - tipo de rotação de arquivo de log e outras opções. Valores possíveis rotate (padrão), copytruncate - observe que copytruncate não pode ser usado junto com maxdelay. Neste caso, o maxdelay deve ser 0 ou não especificado; consulte as notas sobre copytruncate, mtime-reread - registros não únicos, reler se o tempo de modificação ou o tamanho mudar (padrão), mtime-noread - registros não únicos, reler apenas se o tamanho mudar (ignorar alteração no tempo de modificação).</p> <p>persistent_dir (desde versões 5.0.18, 5.4.9, apenas no zabbix_agentd em Unix systems; não suportado no Agent2) - caminho absoluto do diretório onde armazenar arquivos persistentes. Veja também notas adicionais sobre persistent files.</p>	<p>O item deve estar configurado como um active check.</p> <p>As linhas correspondentes são contadas nas novas linhas desde a última verificação de log pelo agente e, portanto, dependem do intervalo de atualização do item.</p> <p>A rotação de log é baseada na última data de modificação dos arquivos.</p> <p>Este item não é suportado para Windows Event Log.</p> <p>Suportado desde a versão 3.2.0 do Zabbix.</p>

Dado Modbus

Chave do item			
Descrição	Valor retornado	Parâmetros	Comentários
modbus.get [endpoint,<slave id>,<function>,<address>,<count>,<type>,<endianness>,<offset>]			

Chave do item			
Reads Modbus data.	JSON object	endpoint - endpoint definido como <code>protocol://connection_string</code> slave id - ID do slave função - função Modbus endereço - endereço do primeiro registro, coil ou input somar - número de registros para ler tipo - tipo de dado endianness - configuração endianness offset - número de registros, iniciando por 'endereço', os resultados que serão descartados. Para mais detalhes sobre os parâmetros, consulte detailed description .	Suportado desde o Zabbix 5.2.0.

Dados de rede

Chave do item			
Descrição	Valor de retorno	Parâmetros	Comentários
net.dns [<ip>,name,<type>,<timeout>,<count>,<protocol>]			
Verifica se o serviço DNS está ativo.	0 - DNS está fora do ar (o servidor não respondeu ou a resolução DNS falhou) 1 - DNS está ativo	ip - Endereço IP do servidor DNS (deixe vazio para o servidor DNS padrão, no Windows é suportado apenas pelo Zabbix agent 2, ignorado pelo Zabbix agent) name - Nome DNS a ser consultado type - Tipo de registro a ser consultado (o padrão é SOA) timeout (ignorado no Windows, a menos que esteja usando a versão 6.0.1 ou mais recente do Zabbix agent 2) - tempo limite para a solicitação em segundos (o padrão é 1 segundo) count (ignorado no Windows, a menos que esteja usando o Zabbix agent 2 versão 6.0.1 ou mais recente) - número de tentativas para a solicitação (o padrão é 2) protocol (desde a versão 3.0) - o protocolo usado para realizar consultas DNS: udp (padrão) ou tcp	Exemplo: => <code>net.dns[8.8.8.8,example.com,MX,2,1]</code> Os valores possíveis são type: ANY, A, NS, CNAME, MB, MG, MR, PTR, MD, MF, MX, SOA, NULL, WKS (não suportado no agente Zabbix no Windows, Zabbix agent 2 em todos OS), HINFO, MINFO, TXT, AAAA, SRV Nomes de domínio internacionalizados não são suportados, por favor, use nomes codificados em IDNA. O tipo de registro SRV é suportado desde o Zabbix 1.8.6 (Unix) e 2.0.0 (Windows). Nomeação antes do Zabbix 2.0 (ainda suportado): <code>net.tcp.dns</code>
net.dns.record [<ip>,name,<type>,<timeout>,<count>,<protocol>]			

Chave do item			
Realiza uma consulta DNS.	String de caracteres com o tipo de informação requerido	<p>ip - Endereço IP do servidor DNS (deixe vazio para o servidor DNS padrão, ignorado no Windows, a menos que esteja usando a versão 6.0.1 ou mais recente do Zabbix agent 2)</p> <p>name - Nome DNS a ser consultado</p> <p>type - Tipo de registro a ser consultado (padrão é SOA_)</p> <p>timeout (ignorado no Windows, a menos que esteja usando a versão 6.0.1 ou mais recente do Zabbix agent 2) - tempo limite para a solicitação em segundos (padrão é 1 segundo)</p> <p>count (ignorado no Windows, a menos que esteja usando a versão 6.0.1 ou mais recente do Zabbix agent 2) - número de tentativas para a solicitação (padrão é 2)</p> <p>protocol(desde a versão 3.0) - o protocolo usado para realizar consultas DNS: udp (padrão) ou tcp</p>	<p>Por exemplo:</p> <pre>=> net.dns.record[8.8.8.8,example.com,MX,2,1]</pre> <p>Os valores possíveis são type: ANY, A, NS, CNAME, MB, MG, MR, PTR, MD, MF, MX, SOA, NULL, WKS (não suportado para Zabbix agent no Windows, Zabbix agent 2 em todos os OS), HINFO, MINFO, TXT, AAAA, SRV</p> <p>Nomes de domínio internacionalizados não são suportados, use nomes codificados em IDNA.</p> <p>SO tipo de registro SRV é suportado desde o Zabbix 1.8.6 (Unix) e 2.0.0 (Windows).</p> <p>Nomeação antes do Zabbix 2.0 (ainda suportado): net.tcp.dns.query</p>
<p>net.if.collisions[if]</p> <p>Número de colisões fora da janela.</p> <p>net.if.discovery</p> <p>Lista de interfaces de rede. Usado para descoberta de baixo nível.</p>	<p>Inteiro</p> <p>Objeto JSON</p>	<p>if - nome da interface de rede</p>	<p>Suportado desde o Zabbix 2.0</p> <p>Em FreeBSD, OpenBSD e NetBSD suportado desde o Zabbix 2.2.</p> <p>Algumas versões do Windows (por exemplo, Server 2008) podem exigir as atualizações mais recentes instaladas para suportar caracteres não ASCII nos nomes das interfaces.</p>
<p>net.if.in[if,<mode>]</p> <p>Estatísticas de tráfego de entrada na interface de rede.</p>	<p>Inteiro</p>	<p>if - nome da interface de rede (Unix); descrição completa da interface de rede ou endereço IPv4; ou, se em colchetes, GUID da interface de rede (Windows)</p> <p>mode - valores possíveis:</p> <p>bytes - número de bytes (padrão)</p> <p>packets - número de pacotes</p> <p>errors - número de erros</p> <p>dropped - número de pacotes descartados</p> <p>overruns (fifo) - número de erros no buffer FIFO</p> <p>frame - número de erros de enquadramento de pacotes</p> <p>compressed - número de pacotes comprimidos recebidos pelo driver do dispositivo</p> <p>multicast - número de frames multicast recebidos pelo driver do dispositivo</p>	<p>No Windows, o item obtém valores de contadores de 64 bits, se disponíveis. Contadores estatísticos de interface de 64 bits foram introduzidos no Windows Vista e Windows Server 2008. Se contadores de 64 bits não estiverem disponíveis, o agente usa contadores de 32 bits.</p> <p>Nomes de interface multibyte no Windows são suportados desde o Zabbix 1.8.6.</p> <p>Exemplos:</p> <pre>=> net.if.in[eth0,errors] => net.if.in[eth0]</pre> <p>Você pode obter descrições de interfaces de rede no Windows com os itens net.if.discovery ou net.if.list items.</p> <p>Você pode usar esta chave com a etapa de pré-processamento Change per second para obter estatísticas de bytes por segundo.</p>
net.if.out [if,<mode>]			

Chave do item			
Estatísticas de tráfego de saída na interface de rede.	Inteiro	<p>if - nome da interface de rede (Unix); descrição completa da interface de rede ou endereço IPv4; ou, se entre chaves, GUID da interface de rede (Windows)</p> <p>mode - Valores possíveis:</p> <p>bytes - número de bytes (padrão)</p> <p>packets - número de pacotes</p> <p>errors - número de erros</p> <p>dropped - número de pacotes descartados</p> <p>overruns (fifo) - número de erros no buffer FIFO</p> <p>collisions (colls) - número de colisões detectadas na interface</p> <p>carrier - número de perdas de porta detectadas pelo driver do dispositivo</p> <p>compressed - número de pacotes comprimidos transmitidos pelo driver do dispositivo</p>	<p>No Windows, o item obtém valores de contadores de 64 bits, se disponíveis. Contadores de estatísticas de interface de 64 bits foram introduzidos no Windows Vista e no Windows Server 2008. Se contadores de 64 bits não estiverem disponíveis, o agente usa contadores de 32 bits.</p> <p>Nomes de interface multi-byte no Windows são suportados desde o Zabbix 1.8.6.</p> <p>Exemplos:</p> <pre>=> net.if.out[eth0,errors] => net.if.out[eth0]</pre> <p>Você pode obter descrições de interfaces de rede no Windows com os itens <code>net.if.discovery</code> ou <code>net.if.list</code>.</p> <p>Você pode usar esta chave com a etapa de pré-processamento <code>Change per second</code> para obter estatísticas de bytes por segundo.</p>
net.if.total [if,<mode>]			
Soma das estatísticas de tráfego de entrada e saída na interface de rede.	Inteiro	<p>if - nome da interface de rede (Unix); descrição completa da interface de rede ou endereço IPv4; ou, se entre chaves, GUID da interface de rede (Windows)</p> <p>mode - Valores possíveis:</p> <p>bytes - número de bytes (padrão)</p> <p>packets - número de pacotes</p> <p>errors - número de erros</p> <p>dropped - número de pacotes descartados</p> <p>overruns (fifo) - número de erros de buffer FIFO</p> <p>compressed - número de colisões detectadas na interface</p>	<p>No Windows, o item obtém valores de contadores de 64 bits, se disponíveis. Contadores de estatísticas de interface de 64 bits foram introduzidos no Windows Vista e no Windows Server 2008. Se os contadores de 64 bits não estiverem disponíveis, o agente usa contadores de 32 bits.</p> <p>Exemplos:</p> <pre>=> net.if.total[eth0,errors] => net.if.total[eth0]</pre> <p>Você pode obter descrições das interfaces de rede no Windows com os itens ou <code>net.if.list</code></p> <p>Você pode usar esta chave com a etapa de pré-processamento <code>Change per second</code> para obter estatísticas de bytes por segundo.</p> <p>Observe que pacotes descartados são suportados apenas se tanto <code>net.if.in</code> quanto <code>net.if.out</code> funcionarem para pacotes descartados em sua plataforma.</p>
net.tcp.listen [port]			

Chave do item			
Verifica se esta porta TCP está no estado LISTEN.	0 - não está no estado LISTEN 1 - está no estado LISTEN	port - número da porta TCP	<p>Exemplo: => net.tcp.listen[80]</p> <p>No Linux é suportado desde o Zabbix 1.8.4</p> <p>Desde o Zabbix 3.0.0, nos kernels Linux 2.6.14 e acima, informações sobre sockets TCP em escuta são obtidas da interface NETLINK do kernel, se possível. Caso contrário, as informações são obtidas dos arquivos /proc/net/tcp e /proc/net/tcp6.</p>
net.tcp.port [<ip>,<port>]			
Verifica se é possível fazer uma conexão TCP para a porta especificada.	0 - não é possível conectar 1 - é possível conectar	<p>ip - Endereço de IP ou nome DNS (padrão 127.0.0.1)</p> <p>port - número da porta</p>	<p>Exemplo: => net.tcp.port[,80] → pode ser usado para testar a disponibilidade do servidor web que está executando na porta.</p> <p>Para testes simples de desempenho TCP, use net.tcp.service.perf[tcp,<ip>,<port>]</p> <p>Observe que essas verificações podem resultar em mensagens adicionais nos arquivos de log do sistema daemon (sessões SMTP e SSH geralmente são registradas).</p>
net.tcp.service [service,<ip>,<port>]			
Verifica se o serviço está em execução e aceitando conexões.	0 - serviço está indisponível 1 - serviço está disponível	<p>service - um dos seguintes: ssh, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet (consulte details)</p> <p>ip - endereço IP ou nome DNS (padrão é 127.0.0.1)</p> <p>port - número da porta (número da porta (por padrão, é utilizado o número da porta padrão do serviço)</p>	<p>Exemplo: => net.tcp.service[ftp,,45] → pode ser usado para testar a disponibilidade do servidor FTP na porta TCP 45.</p> <p>Observe que essas verificações podem resultar em mensagens adicionais nos arquivos de log do sistema daemon (sessões SMTP e SSH geralmente são registradas).</p> <p>A verificação de protocolos criptografados (como IMAP na porta 993 ou POP na porta 995) não é suportada atualmente. Como solução alternativa, use para verificações como essas net.tcp.port.</p> <p>A verificação de LDAP e HTTPS no Windows é suportada apenas pelo Zabbix agent 2.</p> <p>Observe que a verificação do telnet procura por um prompt de login (':' no final).</p> <p>Os serviços https e telnet são suportados desde o Zabbix 2.0.</p>
net.tcp.service.perf [service,<ip>,<port>]			

Chave do item			
Verifica o desempenho do serviço TCP.	Float 0 - serviço está indisponível segundos - o número de segundos gastos enquanto se conecta ao serviço	service - um dos seguintes: ssh, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet (consulte details) ip - Endereço IP ou nome DNS (padrão é 127.0.0.1) port - número da porta (por padrão, é utilizado o número da porta padrão do serviço)	Exemplo: => net.tcp.service.perf[ssh] → pode ser usado para testar a velocidade da resposta inicial do servidor SSH. A verificação de protocolos criptografados (como IMAP na porta 993 ou POP na porta 995) não é suportada atualmente. Como solução alternativa, use net.tcp.service.perf[tcp,<ip>,<port>] para verificações como essas. A verificação de LDAP e HTTPS no Windows é suportada apenas pelo Zabbix agent 2. Observe que a verificação do telnet procura por um prompt de login (':' no final). Os serviços https e telnet são suportados desde o Zabbix 2.0.
net.tcp.socket.count [<laddr>,<lport>,<raddr>,<rport>,<state>]	Retorne o número de sockets TCP que correspondem aos parâmetros.	Inteiro laddr - endereço IPv4/6 local ou sub-rede CIDR lport - número da porta local ou nome do serviço raddr - endereço IPv4/6 remoto ou sub-rede CIDR rport - número da porta remota ou nome do serviço state - estado da conexão (established, syn_sent, syn_recv, fin_wait1, fin_wait2, time_wait, close, close_wait, last_ack, listen, closing)	Este item é suportado no Linux apenas em ambos os Zabbix agent/agent 2. No Zabbix agent 2, também é suportado no Windows de 64 bits. Exemplo: => net.tcp.socket.count[,80,,established] → retorna o número de conexões para a porta TCP local 80 no estado estabelecido Esse item é suportado desde o Zabbix 6.0.
net.udp.listen [port]	Verifica se esta porta UDP está no estado LISTEN.	0 - não está no estado LISTEN 1 - está no estado LISTEN	Exemplo: => net.udp.listen[68] No Linux suportado desde o Zabbix 1.8.4
net.udp.service [service,<ip>,<port>]	Verifica se o serviço está em execução e respondendo a solicitações UDP.	0 - serviço está indisponível 1 - serviço está disponível service - ntp (consulte details) ip - nome do IP ou DNS (padrão é 127.0.0.1) port - número da porta por padrão, o número da porta do serviço padrão é usado	Exemplo: => net.udp.service[ntp,,45] → pode ser usado para testar a disponibilidade do serviço NTP na porta UDP 45. Esse item é suportado desde Zabbix 3.0.0, mas o serviço ntp estava disponível para item .tcp.service[] nas versões anteriores.
net.udp.service.perf [service,<ip>,<port>]			

Chave do item			
Verifica o desempenho do serviço UDP.	Float 0 - serviço está indisponível segundos - número de segundos gastos esperando uma resposta do serviço	service - ntp (consulte details) ip - Nome do IP ou DNS (padrão é 127.0.0.1) port - número da porta (by padrão o número da porta de serviço é usado)	Exemplo: => net.udp.service.perf[ntp] → pode ser usado para testar o tempo de resposta do serviço NTP. Esse item é suportado desde o Zabbix 3.0.0, mas o serviço ntp estava disponível para item net.tcp.service[] nas versões anteriores.
net.udp.socket.count [<laddr>,<lport>,<raddr>,<rport>,<state>]			
Retorna o número de sockets UDP que correspondem aos parâmetros.	Inteiro	laddr - endereço IPv4/6 local ou sub-rede lport - número da porta local ou nome do serviço raddr - remote IPv4/6 address or CIDR subnet rport - número da porta remota ou nome do serviço state - estado da conexão (established,unconn)	Este item é suportado apenas no Linux tanto no Zabbix agent/agent 2. No Zabbix agent 2 também é suportado no Windows 64-bit. Exemplo: => net.udp.socket.count[,,,established] → retorna o número de sockets UDP no estado conectado Este item é suportado desde o Zabbix 6.0.

Processes

Key	Description	Return value	Parameters	Comments
proc.cpu.util[<name>,<user>,<type>,<cmdline>,<mode>,<zone>]				

Process CPU
utilization
percentage.

Float

name -
process name
(default is all
processes)
user - user
name (default
is all users)
type - CPU
utilization type:
total (default),
user, system
cmdline - filter
by command
line (it is a
regular
expression)
mode - data
gathering
mode: avg1
(default), avg5,
avg15
zone - target
zone: current
(default), all.
This parameter
is supported on
Solaris only.

Examples:

=>
proc.cpu.util[,root]
→ CPU
utilization of all
processes
running under
the "root" user
=>
proc.cpu.util[zabbix_server,za]
→ CPU
utilization of all
zabbix_server
processes
running under
the zabbix user

The returned
value is based
on single CPU
core utilization
percentage.
For example
CPU utilization
of a process
fully using two
cores is 200%.

The process
CPU utilization
data is
gathered by a
collector which
supports the
maximum of
1024 unique
(by name, user
and command
line) queries.
Queries not
accessed
during the last
24 hours are
removed from
the collector.

Note that when
setting the
zone
parameter to
current (or
default) in case
the agent has
been compiled
on a Solaris
without zone
support, but
running on a
newer Solaris
where zones
are supported,
then the agent
will return NOT-
SUPPORTED
(the agent

Key

proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]

Memory used by process in bytes.	Integer - with mode as max, min, sum Float - with mode as avg	name - process name (default is all processes) user - user name (default is all users) mode - possible values: avg, max, min, sum (default) cmdline - filter by command line (it is a regular expression) memtype - type of memory used by process	<p>Examples:</p> <p>=> proc.mem[,root] → memory used by all processes running under the "root" user</p> <p>=> proc.mem[zabbix_server,zabbix_server] → memory used by all zabbix_server processes running under the zabbix user</p> <p>=> proc.mem[,oracle,max,oracle2] → memory used by the most memory-hungry process running under oracle having oracleZABBIX in its command line</p> <p>Note: When several processes use shared memory, the sum of memory used by processes may result in large, unrealistic values.</p> <p>See notes on selecting processes with name and cmdline parameters (Linux-specific).</p> <p>When this item is invoked from the command line and contains a command line parameter (e.g. using the agent test mode: zabbix_agentd -t proc.mem[, , ,apache2]), one extra process will be</p>
----------------------------------	--	---	---

Key

proc.num[<name>,<user>,<state>,<cmdline>,<zone>]

The number of
processes.

Integer

name -
process name
(default is all
processes)
user - user
name (default
is all users)
state (disk and
trace options
since version
3.4.0) -
possible
values:
all (default),
disk -
uninterruptible
sleep,
run - running,
sleep -
interruptible
sleep,
trace -
stopped,
zomb - zombie
cmdline - filter
by command
line (it is a
regular
expression)
zone - target
zone: current
(default), all.
This parameter
is supported on
Solaris only.

Examples:

=>
proc.num[,mysql]
→ number of
processes
running under
the mysql user
=>
proc.num[apache2,www-
data] →
number of
apache2
processes
running under
the www-data
user
=>
proc.num[,oracle,sleep,oracle]
→ number of
processes in
sleep state
running under
oracle having
oracleZABBIX
in its command
line

See [notes](#) on
selecting
processes with
name and
cmdline
parameters
(Linux-
specific).

On Windows,
only the **name**
and **user**
parameters are
supported.

When this item
is invoked from
the command
line and
contains a
command line
parameter
(e.g. using the
agent test
mode:
zabbix_agentd
-t
proc.num[, , ,apache2]),
one extra
process will be
counted, as the
agent will
count itself.

Note that when
setting the
zone
parameter to

Key

Sensors

Key

	Description	Return value	Parameters	Comments
sensor[device,sensor,<mode>]	Hardware sensor reading.	Float	device - device name sensor - sensor name mode - possible values: avg, max, min (if this parameter is omitted, device and sensor are treated verbatim).	<p>Reads /proc/sys/dev/sensors on Linux 2.4.</p> <p>Example: => sensor[w83781d-i2c-0-2d,temp1]</p> <p>Prior to Zabbix 1.8.4, the sensor[temp1] format was used.</p> <p>Reads /sys/class/hwmon on Linux 2.6+.</p> <p>See a more detailed description of sensor item on Linux.</p> <p>Reads the hw.sensors MIB on OpenBSD.</p> <p>Examples: => sensor[cpu0,temp0] → temperature of one CPU => sensor["cpu[0-2]\$",temp,avg] → average temperature of the first three CPU's</p> <p>Supported on OpenBSD since Zabbix 1.8.4.</p>

System data

Key

	Description	Return value	Parameters	Comments
system.boottime	System boot time.	Integer (Unix timestamp)		
system.cpu.discovery				

Key				
	List of detected CPUs/CPU cores. Used for low-level discovery.	JSON object		Supported on all platforms since 2.4.0.
system.cpu.intr	Device interrupts.	Integer		
system.cpu.load[<cpu>,<mode>]	CPU load.	Float	cpu - possible values: all (default), percpu (since version 2.0; total load divided by online CPU count) mode - possible values: avg1 (one-minute average, default), avg5, avg15	Example: => sys-tem.cpu.load[,avg5].
system.cpu.num[<type>]	Number of CPUs.	Integer	type - possible values: online (default), max	Example: => sys-tem.cpu.num
system.cpu.switches	Count of context switches.	Integer		
system.cpu.util[<cpu>,<type>,<mode>,<logical_or_physical>]				

Key				
	CPU utilization percentage.	Float	<p>cpu - <CPU number> or all (default)</p> <p>type - possible values: user (default), idle, nice, system (default for Windows), iowait, interrupt, softirq, steal, guest (on Linux kernels 2.6.24 and above), guest_nice (on Linux kernels 2.6.33 and above). See also platform-specific details for this parameter.</p> <p>mode - possible values: avg1 (one-minute average, default), avg5, avg15</p> <p>logical_or_physical (since version 5.0.3; on AIX only) - possible values: logical (default), physical. This parameter is supported on AIX only.</p>	<p>Example: => sys-tem.cpu.util[0,user,avg5]</p> <p>Old naming: sys-tem.cpu.idleX, sys-tem.cpu.niceX, sys-tem.cpu.systemX, sys-tem.cpu.userX</p>
system.hostname[<type>, <transform>]				

Key	System host name.	String	type (before version 5.4.7 supported on Windows only) - possible values: netbios (default on Windows), host (default on Linux), shorthost (since version 5.4.7; returns part of the hostname before the first dot, a full string for names without dots). transform (since version 5.4.7) - possible values: none (default), lower (convert to lowercase)	The value is acquired by either GetComputerName() (for netbios) or gethostname() (for host) functions on Windows and by "hostname" command on other systems. Examples of returned values: on Linux: => sys-tem.hostname → linux-w7x1 => sys-tem.hostname → example.com => sys-tem.hostname[shorthost] → example on Windows: => sys-tem.hostname → WIN-SERV2008-I6 => sys-tem.hostname[host] → Win-Serv2008-I6LonG => sys-tem.hostname[host,lower] → win-serv2008-i6long See also a more detailed description .
system.hw.chassis[<info>]				

Key				
	Chassis information.	String	info - one of full (default), model, serial, type or vendor	<p>Example: system.hw.chassis[full] Hewlett-Packard HP Pro 3010 Small Form Factor PC CZXXXXXXXX Desktop]</p> <p>This key depends on the availability of the SMBIOS table. Will try to read the DMI table from sysfs, if sysfs access fails then try reading directly from memory.</p> <p>Root permissions are required because the value is acquired by reading from sysfs or memory.</p> <p>Supported since Zabbix 2.0.</p>
system.hw.cpu[<cpu>,<info>]	CPU information.	String or integer	<p>cpu - <CPU number> or all (default)</p> <p>info - possible values: full (default), curfreq, maxfreq, model or vendor</p>	<p>Example: => system.hw.cpu[0,vendor] → AuthenticAMD</p> <p>Gathers info from /proc/cpuinfo and /sys/devices/system/cpu/[cpu</p> <p>If a CPU number and curfreq or maxfreq is specified, a numeric value is returned (Hz).</p> <p>Supported since Zabbix 2.0.</p>

Key

system.hw.devices[<type>]

Listing of PCI or
USB devices.

Text

type (since
version 2.0) -
pci (default) or
usb

Example:
=> sys-
tem.hw.devices[pci]
→ 00:00.0 Host
bridge:
Advanced
Micro Devices
[AMD] RS780
Host Bridge
[..]

Returns the
output of
either lspci or
lsusb utility
(executed
without any
parameters).

system.hw.macaddr[<interface>,<format>]

Listing of MAC
addresses.

String

interface - all
(default) or a
regular
expression
format - full
(default) or
short

Lists MAC
addresses of
the interfaces
whose name
matches the
given
interface
regular
expression (all
lists for all
interfaces).

Example:
=> sys-
tem.hw.macaddr["eth0\$",full]
→ [eth0]
00:11:22:33:44:55

If format is
specified as
short, interface
names and
identical MAC
addresses are
not listed.

Supported
since Zabbix
2.0.

system.localtime[<type>]

Key

System time.

Integer - with
type as utc

String - with
type as local

type (since
version 2.0) -
possible
values:
utc - (default)
the time since
the Epoch
(00:00:00 UTC,
January 1,
1970),
measured in
seconds.
local - the time
in the
'yyyy-mm-
dd,hh:mm:ss.nnn,
format

Must be used
as a **passive**
check only.

Example:
=> sys-
tem.localtime[local]
→ create an
item using this
key and then
use it to
display host
time in the
Clock
dashboard
widget

system.run[command,<mode>]

	Run specified command on the host.	Text result of the command 1 - with mode as nowait (regardless of command result)	command - command for execution mode - possible values: wait - wait end of execution (default), nowait - do not wait	Up to 512KB of data can be returned, including trailing whitespace that is truncated. To be processed correctly, the output of the command must be text. Example: => system.run[ls -l /] → detailed file list of root directory. Note: system.run items are disabled by default. Learn how to enable them . The return value of the item is standard output together with standard error produced by command. The exit code is not checked. Empty result is allowed starting with Zabbix 2.4.0. See also: Command execution .
system.stat[resource,<type>]				

System statistics.	Integer or float	<p>ent - number of processor units this partition is entitled to receive (float)</p> <p>kthr,<type> - information about kernel thread states:</p> <p>r - average number of runnable kernel threads (float)</p> <p>b - average number of kernel threads placed in the Virtual Memory Manager wait queue (float)</p> <p>memory,<type> - information about the usage of virtual and real memory:</p> <p>avm - active virtual pages (integer)</p> <p>fre - size of the free list (integer)</p> <p>page,<type> - information about page faults and paging activity:</p> <p>fi - file page-ins per second (float)</p> <p>fo - file page-outs per second (float)</p> <p>pi - pages paged in from paging space (float)</p> <p>po - pages paged out to paging space (float)</p> <p>fr - pages freed (page replacement) (float)</p> <p>sr - pages scanned by page-replacement algorithm (float)</p> <p>faults,<type> - trap and</p>
--------------------	------------------	--

				Comments
				<p>This item is supported on AIX only, since Zabbix 1.8.1.</p> <p>Take note of the following limitations in these items:</p> <p>=> sys-tem.stat[cpu,app]</p> <p>- supported only on AIX LPAR of type "Shared"</p> <p>=> sys-tem.stat[cpu,ec]</p> <p>- supported on AIX LPAR of type "Shared" and "Dedicated" ("Dedicated" always returns 100 (percent))</p> <p>=> sys-tem.stat[cpu,lbusy]</p> <p>- supported only on AIX LPAR of type "Shared"</p> <p>=> sys-tem.stat[cpu,pc]</p> <p>- supported on AIX LPAR of type "Shared" and "Dedicated"</p> <p>=> sys-tem.stat[ent] - supported on AIX LPAR of type "Shared" and "Dedicated"</p>
system.sw.arch	Software architecture information.	String		<p>Example:</p> <p>=> system.sw.arch → i686</p> <p>Info is acquired from uname() function.</p> <p>Supported since Zabbix 2.0.</p>
system.sw.os[<info>]				

Key				
	Operating system information.	String	info - possible values: full (default), short or name	<p>Example: => sys-tem.sw.os[short]→ Ubuntu 2.6.35-28.50-generic 2.6.35.11</p> <p>Info is acquired from (note that not all files and options are present in all distributions): /proc/version (full) /proc/version_signature (short) PRETTY_NAME parameter from /etc/os-release on systems supporting it, or /etc/issue.net (name)</p> <p>Supported since Zabbix 2.0.</p>
system.sw.packages[<package>,<manager>,<format>]				

Key				
	Listing of installed packages.	Text	package - all (default) or a regular expression manager - all (default) or a package manager format - full (default) or short	<p>Lists (alphabetically) installed packages whose name matches the given package regular expression (all lists them all).</p> <p>Example: => system.sw.packages[mini,dpkg,sl → python-minimal, python2.6-minimal, ubuntu-minimal</p> <p>Supported package managers (executed command): dpkg (dpkg --get-selections) pkgtool (ls /var/log/packages) rpm (rpm -qa) pacman (pacman -Q)</p> <p>If format is specified as full, packages are grouped by package managers (each manager on a separate line beginning with its name in square brackets). If format is specified as short, packages are not grouped and are listed on a single line.</p> <p>Supported since Zabbix 2.0.</p>
system.swap.in[<device>,<type>]				

	Swap in (from device into memory) statistics.	Integer	device - device used for swapping (default is all) type - possible values: count (number of swapins), sectors (sectors swapped in), pages (pages swapped in). See also platform-specific details for this parameter.	Example: => sys-tem.swap.in[,pages] The source of this information is: /proc/swaps, /proc/partitions, /proc/stat (Linux 2.4) /proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)
system.swap.out[<device>,<type>]	Swap out (from memory onto device) statistics.	Integer	device - device used for swapping (default is all) type - possible values: count (number of swapouts), sectors (sectors swapped out), pages (pages swapped out). See also platform-specific details for this parameter.	Example: => sys-tem.swap.out[,pages] The source of this information is: /proc/swaps, /proc/partitions, /proc/stat (Linux 2.4) /proc/swaps, /proc/diskstats, /proc/vmstat (Linux 2.6)
system.swap.size[<device>,<type>]				

	Swap space size in bytes or in percentage from total.	Integer - for bytes Float - for percentage	device - device used for swapping (default is all) type - possible values: free (free swap space, default), pfree (free swap space, in percent), pused (used swap space, in percent), total (total swap space), used (used swap space) Note that pfree, pused are not supported on Windows if swap size is 0. See also platform-specific details for this parameter.	Example: => sys-tem.swap.size[,pfree] → free swap space percentage If device is not specified Zabbix agent will only take into account swap devices (files), physical memory will be ignored. For example, on Solaris systems swap -s command includes a portion of physical memory and swap devices (unlike swap -l). Note that this key might report incorrect swap space size/percentage on virtualized (VMware ESXi, VirtualBox) Windows platforms. In this case you may use the perf_counter[\700(_Total) key to obtain correct swap space percentage.
system.uname				

Identification of the system.	String	Example of returned value (Unix): FreeBSD localhost 4.2-RELEASE FreeBSD 4.2-RELEASE #0: Mon Nov i386 Example of returned value (Windows): Windows ZABBIX-WIN 6.0.6001 Microsoft® Windows Server® 2008 Standard Service Pack 1 x86 On Unix since Zabbix 2.2.0 the value for this item is obtained with uname() system call. Previously it was obtained by invoking "uname -a". The value of this item might differ from the output of "uname -a" and does not include additional information that "uname -a" prints based on other sources. On Windows since Zabbix 3.0 the value for this item is obtained from Win32_OperatingSystem and Win32_Processor WMI classes. Previously it was obtained from volatile Windows APIs and undocumented registry keys.
-------------------------------	--------	---

Key			
system.uptime	System uptime in seconds.	Integer	In item configuration , use s or uptime units to get readable values.
system.users.num	Number of users logged in.	Integer	who command is used on the agent side to obtain the value.

Virtual file systems

Key				
	Description	Return value	Parameters	Comments
vfs.dev.discovery	List of block devices and their type. Used for low-level discovery.	JSON object		This item is supported on Linux platform only. Supported since Zabbix 4.4.0.
vfs.dev.read[<device>,<type>,<mode>]				

	Disk read statistics.	<p>Integer - with type in sectors, operations, bytes</p> <p>Float - with type in sps, ops, bps</p> <p>Note: if using an update interval of three hours or more², will always return '0'</p>	<p>device - disk device (default is all³)</p> <p>type - possible values: sectors, operations, bytes, sps, ops, bps</p> <p>Note that 'type' parameter support and defaults depend on the platform. See platform-specific details. sps, ops, bps stand for: sectors, operations, bytes per second, respectively.</p> <p>mode - possible values: avg1 (one-minute average, default), avg5, avg15. This parameter is supported only with type in: sps, ops, bps.</p>	<p>You may use relative device names (for example, sda) as well as an optional /dev/ prefix (for example, /dev/sda).</p> <p>LVM logical volumes are supported.</p> <p>Default values of 'type' parameter for different OSes: AIX - operations FreeBSD - bps Linux - sps OpenBSD - operations Solaris - bytes</p> <p>Example: => vfs.dev.read[,operations]</p> <p>sps, ops and bps on supported platforms used to be limited to 8 devices (7 individual and one all). Since Zabbix 2.0.1 this limit is 1024 devices (1023 individual and one for all).</p>
--	-----------------------	--	---	--

vfs.dev.write[<device>,<type>,<mode>]

Disk write statistics.	Integer - with type in sectors, operations, bytes Float - with type in sps, ops, bps Note: if using an update interval of three hours or more ² , will always return '0'	device - disk device (default is all ³) type - possible values: sectors, operations, bytes, sps, ops, bps Note that 'type' parameter support and defaults depend on the platform. See platform-specific details. sps, ops, bps stand for: sectors, operations, bytes per second, respectively. mode - possible values: avg1 (one-minute average, default), avg5, avg15. This parameter is supported only with type in: sps, ops, bps.	You may use relative device names (for example, sda) as well as an optional /dev/ prefix (for example, /dev/sda). LVM logical volumes are supported. Default values of 'type' parameter for different OSes: AIX - operations FreeBSD - bps Linux - sps OpenBSD - operations Solaris - bytes Example: => vfs.dev.write[,operations] sps, ops and bps on supported platforms used to be limited to 8 devices (7 individual and one all). Since Zabbix 2.0.1 this limit is 1024 (1023 individual and one for all).
------------------------	---	---	---

vfs.dir.count[dir,<regex_incl>,<regex_excl>,<types_incl>,<types_excl>,<max_depth>,<min_size>,<max_size>,<min_age>,<max_age>]

Directory entry count.	Integer		
		dir - absolute path to directory regex_incl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to include; include all if empty (default value) regex_excl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to exclude; don't exclude any if empty (default value) types_incl - directory entry types to count, possible values: file - regular file, dir - subdirectory, sym - symbolic link, sock - socket, bdev - block device, cdev - character device, fifo - FIFO, dev - synonymous with "bdev,cdev", all - all types (default), i.e. "file,dir,sym,sock,bdev,fifo,dev". Multiple types must be separated with comma and quoted. types_excl - directory entry types (see <types_incl>) to NOT count. If some entry type is in both <types_incl> and <types_excl>, directory	Environment variables, e.g. %APP_HOME%, \$HOME and %TEMP% are not supported. Pseudo-directories "." and ".." are never counted. Symbolic links are never followed for directory traversal. On Windows, directory symlinks are skipped and hard links are counted only once. Both regex_incl and regex_excl are being applied to files and directories when calculating entry size, but are ignored when picking subdirectories to traverse (if regex_incl is "(?i)^.+\\.zip\$" and max_depth is not set, then all subdirectories will be traversed, but only files, not directories, will be counted). Execution time is limited by the default timeout value in agent configuration (3 sec). Since large directory traversal may take longer than that, no data will be returned and

Key

vfs.dir.get[dir,<regex_incl>,<regex_excl>,<types_incl>,<types_excl>,<max_depth>,<min_size>,<max_size>,<min_age>,<max_age>

Directory entry list.	JSON	Environment variables, e.g. %APP_HOME%, \$HOME and %TEMP% are not supported.
dir - absolute path to directory	dir - absolute path to directory	Environment variables, e.g. %APP_HOME%, \$HOME and %TEMP% are not supported.
regex_incl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to include; include all if empty (default value)	regex_incl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to include; include all if empty (default value)	Pseudo-directories "." and ".." are never listed.
regex_excl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to exclude; don't exclude any if empty (default value)	regex_excl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to exclude; don't exclude any if empty (default value)	Symbolic links are never followed for directory traversal.
types_incl - directory entry types to list, possible values: file - regular file, dir - subdirectory, sym - symbolic link, sock - socket, bdev - block device, cdev - character device, fifo - FIFO, dev - synonymous with "bdev,cdev", all - all types (default), i.e. "file,dir,sym,sock,bdev,cdev,fifo". Multiple types must be separated with comma and quoted.	types_incl - directory entry types to list, possible values: file - regular file, dir - subdirectory, sym - symbolic link, sock - socket, bdev - block device, cdev - character device, fifo - FIFO, dev - synonymous with "bdev,cdev", all - all types (default), i.e. "file,dir,sym,sock,bdev,cdev,fifo". Multiple types must be separated with comma and quoted.	On Windows, directory symlinks are skipped and hard links are listed only once.
types_excl - directory entry types (see <types_incl>) to NOT list. If some entry type is in both <types_incl> and <types_excl>, directory	types_excl - directory entry types (see <types_incl>) to NOT list. If some entry type is in both <types_incl> and <types_excl>, directory	Both regex_incl and regex_excl are being applied to files and directories when calculating entry size, but are ignored when picking subdirectories to traverse (if regex_incl is "(?i)^.+.zip\$" and max_depth is not set, then all subdirectories will be traversed, but only files and directories are listed).
		Execution time is limited by the default timeout value in agent configuration (3 sec). Since large directory traversal may take longer than that, no data will be returned and

Key

vfs.dir.size[dir,<regex_incl>,<regex_excl>,<mode>,<max_depth>,<regex_excl_dir>]

Directory size (in bytes).	Integer		
		dir - absolute path to directory regex_incl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to include; include all if empty (default value) regex_excl - regular expression describing the name pattern of the entity (file, directory, symbolic link) to exclude; don't exclude any if empty (default value) mode - possible values: apparent (default) - gets apparent file sizes rather than disk usage (acts as <code>du -sb dir</code>), disk - gets disk usage (acts as <code>du -s -B1 dir</code>). Unlike <code>du</code> command, <code>vfs.dir.size</code> item takes hidden files in account when calculating directory size (acts as <code>du -sb . [^.] * *</code> within dir). max_depth - maximum depth of subdirectories to traverse. -1 (default) - unlimited, 0 - no descending into subdirectories. regex_excl_dir - regular expression describing the	Only directories with at least read permission for zabbix user are calculated. On Windows any symlink is skipped and hard links are taken into account only once. With large directories or slow drives this item may time out due to the Timeout setting in agent and server/proxy configuration files. Increase the timeout values as necessary. Examples: ⇒ <code>vfs.dir.size[/tmp,log]</code> - calculates size of all files in /tmp which contain 'log' ⇒ <code>vfs.dir.size[/tmp,log,^[^.]old\$]</code> - calculates size of all files in /tmp which contain 'log', excluding files containing '.old' The file size limit depends on large file support . Supported since Zabbix 3.4.0.

Key

`vfs.file.cksum[file,<mode>]`

File checksum,
calculated by
the UNIX
cksum
algorithm.

Integer - with
mode as crc32

String - with
mode as md5,
sha256

file - full path
to file
mode - crc32
(default), md5,
sha256

Example:
=>
`vfs.file.cksum[/etc/passwd]`

Example of
returned values
(crc32/md5/sha256
respectively):
675436101
9845acf68b73991eb7fd7ee0d
ae67546e4aac995e5c921042

The file size
limit depends
on **large file**
support.

The mode
parameter is
supported
since Zabbix
6.0.

`vfs.file.contents[file,<encoding>]`

Retrieving
contents of a
file.

Text

file - full path
to file
encoding -
code page
identifier

Returns an
empty string if
the file is
empty or
contains LF/CR
characters
only.

Byte order
mark (BOM) is
excluded from
the output.

Example:
=>
`vfs.file.contents[/etc/passwd]`

This item is
limited to files
no larger than
64 Kbytes.

Supported
since Zabbix
2.0.

`vfs.file.exists[file,<types_incl>,<types_excl>]`

<p>Checks if file exists.</p>	<p>0 - not found</p> <p>1 - file of the specified type exists</p>	<p>file - full path to file</p> <p>types_incl - list of file types to include, possible values: file (regular file, default (if types_excl is not set)), dir (directory), sym (symbolic link), sock (socket), bdev (block device), cdev (character device), fifo (FIFO), dev (synonymous with "bdev,cdev"), all (all mentioned types, default if types_excl is set).</p> <p>types_excl - list of file types to exclude, see types_incl for possible values (by default no types are excluded)</p>	<p>Multiple types must be separated with a comma and the entire set enclosed in quotes "".</p> <p>On Windows the double quotes have to be backslash '\ ' escaped and the whole item key enclosed in double quotes when using the command line utility for calling zabbix_get.exe or agent2.</p> <p>If the same type is in both <types_incl> and <types_excl>, files of this type are excluded.</p> <p>Examples: => vfs.file.exists[/tmp/application => vfs.file.exists[/tmp/application => vfs.file.exists[/tmp/application</p> <p>The file size limit depends on large file support.</p> <p>Note that the item may turn unsupported on Windows if a directory is searched within a non-existing directory, e.g. <code>vfs.file.exists[C:\no\dir,dir]</code> (where 'no' does not exist).</p>
-------------------------------	---	--	---

vfs.file.get[file]

Key

	Return information about a file.	JSON object	file - full path to file	Supported file types on UNIX-like systems: regular file, directory, symbolic link, socket, block device, character device, FIFO Supported file types on Windows: regular file, directory, symbolic link Example: => vfs.file.get[/etc/passwd] → return a JSON with information about the /etc/passwd file (type, user, permissions, SID, uid etc) Supported since Zabbix 6.0.
vfs.file.md5sum[file]	MD5 checksum of file.	Character string (MD5 hash of the file)	file - full path to file	Example: => vfs.file.md5sum[/usr/local/etc/...] Example of returned value: b5052dec b577e0fffd622d6dd... The file size limit (64 MB) for this item was removed in version 1.8.6. The file size limit depends on large file support .
vfs.file.owner[file,<ownertype>,<resulttype>]				

Key				
	Retrieve owner of a file.	Character string	file - full path to file ownertype - user (default) or group (Unix only) resulttype - name (default) or id; for id - return uid/gid on Unix, SID on Windows	Example: => vfs.file.owner[/tmp/zabbix_server.log] → return file owner of /tmp/zabbix_server.log => vfs.file.owner[/tmp/zabbix_server.log, id] → return file owner ID of /tmp/zabbix_server.log Supported since Zabbix 6.0.
vfs.file.permissions[file]	Return a 4-digit string containing the octal number with Unix permissions.	String	file - full path to the file	Not supported on Windows. Example: => vfs.file.permissions[/etc/passwd] → return permissions of /etc/passwd, for example, '0644' Supported since Zabbix 6.0.
vfs.file.regexp[file,regexp,<encoding>,<start line>,<end line>,<output>]				

Find string in a file.	The line containing the matched string, or as specified by the optional output parameter	<p>file - full path to file</p> <p>regexp - regular expression describing the required pattern</p> <p>encoding - code page identifier</p> <p>start line - the number of first line to search (first line of file by default).</p> <p>end line - the number of last line to search (last line of file by default).</p> <p>output - an optional output formatting template. The \0 escape sequence is replaced with the matched part of text (from the first character where match begins until the character where match ends) while an \N (where N=1...9) escape sequence is replaced with Nth matched group (or an empty string if the N exceeds the number of captured groups).</p>	<p>Only the first matching line is returned. An empty string is returned if no line matched the expression.</p> <p>Byte order mark (BOM) is excluded from the output.</p> <p>Content extraction using the output parameter takes place on the agent.</p> <p>The start line, end line and output parameters are supported from version 2.2.</p> <p>Examples: => vfs.file.regexp[/etc/passwd,zab => vfs.file.regexp[/path/to/some/f 9]+)\$",,3,5,\1] => vfs.file.regexp[/etc/passwd,"^ 9]+)"",,\1] → getting the ID of user zabbix</p>
------------------------	--	---	---

vfs.file.regmatch[file,regexp,<encoding>,<start line>,<end line>]

Key

	Find string in a file.	0 - match not found 1 - found	file - full path to file regex - regular expression describing the required pattern encoding - code page identifier start line - the number of first line to search (first line of file by default). end line - the number of last line to search (last line of file by default).	Byte order mark (BOM) is ignored. The start line and end line parameters are supported from version 2.2. Example: => vfs.file.regmatch[/var/log/app.
vfs.file.size[file,<mode>]	File size (in bytes).	Integer	file - full path to file mode - possible values: bytes (default) or lines (empty lines are counted, too)	The file must have read permissions for user zabbix. Example: => vfs.file.size[/var/log/syslog] The file size limit depends on large file support . The mode parameter is supported since Zabbix 6.0.
vfs.file.time[file,<mode>]	File time information.	Integer (Unix timestamp)	file - full path to the file mode - possible values: modify (default) - last time of modifying file content, access - last time of reading file, change - last time of changing file properties	Example: => vfs.file.time[/etc/passwd,modi The file size limit depends on large file support .
vfs.fs.discovery				

	List of mounted filesystems and their types. Used for low-level discovery.	JSON object		Supported since Zabbix 2.0. The {#FSDRIVETYPE} macro is supported on Windows since Zabbix 3.0. The {#FSLABEL} macro is supported on Windows since Zabbix 6.0.
vfs.fs.get				
	List of mounted filesystems, their types, disk space and inode statistics. Can be used for low-level discovery.	JSON object		Supported since Zabbix 4.4.5. The {#FSLABEL} macro is supported on Windows since Zabbix 6.0.
vfs.fs.inode[fs,<mode>]	Number or percentage of inodes.	Integer - for number Float - for percentage	fs - filesystem mode - possible values: total (default), free, used, //pfree // (free, percentage), pused (used, percentage)	Example: => vfs.fs.inode[/,pfree]
vfs.fs.size[fs,<mode>]	Disk space in bytes or in percentage from total.	Integer - for bytes Float - for percentage	fs - filesystem mode - possible values: total (default), free, used, pfree (free, percentage), pused (used, percentage)	In case of a mounted volume, disk space for local file system is returned. Example: => vfs.fs.size[/tmp,free] Reserved space of a file system is taken into account and not included when using the free mode.

Key

	Description	Return value	Parameters	Comments
vm.memory.size[<mode>]	Memory size in bytes or in percentage from total.	Integer - for bytes Float - for percentage	mode - possible values: total (default), active, anon, buffers, cached, exec, file, free, inactive, pinned, shared, slab, wired, used, pused (used, percentage), available, pavailable (available, percentage) See also platform-specific support and additional details for this parameter.	This item accepts three categories of parameters: 1) total - total amount of memory; 2) platform-specific memory types: active, anon, buffers, cached, exec, file, free, inactive, pinned, shared, slab, wired; 3) user-level estimates on how much memory is used and available: used, pused, available, pavailable.

Web monitoring

Key

	Description	Return value	Parameters	Comments
web.page.get[host,<path>,<port>]				

Get content of web page.	Web page source as text (including headers)	<p>host - hostname or URL (as <code>scheme://host:specified path</code>, where only host is mandatory). Allowed URL schemes: http, https⁴. Missing scheme will be treated as http. If URL is specified <code>path</code> and <code>port</code> must be empty. Specifying user name/password when connecting to servers that require authentication, for example: <code>http://user:password@http://www.example.com/</code> is only possible with cURL support⁴. Punycode is supported in hostnames.</p> <p>path - path to HTML document (default is /)</p> <p>port - port number (default is 80 for HTTP)</p>	<p>This item turns unsupported if the resource <code>specified path</code>, host does not exist or is unavailable.</p> <p>host can be hostname, domain name, IPv4 or IPv6 address. But for IPv6 address Zabbix agent must be compiled with IPv6 support enabled.</p> <p>Example: => <code>web.page.get[www.example.com]</code> => <code>web.page.get[http://www.example.com/]</code> => <code>web.page.get[https://blog.example.com/]</code> => <code>web.page.get[localhost:80]</code> => <code>web.page.get["::1]/server-status"]</code></p>
--------------------------	---	--	---

web.page.perf[host,<path>,<port>]

Key				
	Loading time of full web page (in seconds).	Float	host - hostname or URL (as scheme://host:port/path, where only host is mandatory). Allowed URL schemes: http, https ⁴ . Missing scheme will be treated as http. If URL is specified path and port must be empty. Specifying user name/password when connecting to servers that require authentication, for example: http://user:password@www.example.com/ is only possible with cURL support ⁴ . Punycode is supported in hostnames. path - path to HTML document (default is /) port - port number (default is 80 for HTTP)	This item turns unsupported if the resource specified path, host does not exist or is unavailable. host can be hostname, domain name, IPv4 or IPv6 address. But for IPv6 address Zabbix agent must be compiled with IPv6 support enabled. Example: => web.page.perf[www.example.com] => web.page.perf[https://www.example.com/]
web.page.regexp[host,<path>,<port>,regexp,<length>,<output>]				

Find string on a web page.	The matched string, or as specified by the optional output parameter	<p>host - hostname or URL (as <code>scheme://host:port/path</code>, where only host is mandatory). Allowed URL schemes: http, https⁴. Missing scheme will be treated as http. If URL is specified path and port must be empty. Specifying user name/password when connecting to servers that require authentication, for example: <code>http://user:password@www.example.com</code> is only possible with cURL support⁴. Punycode is supported in hostnames.</p> <p>path - path to HTML document (default is /)</p> <p>port - port number (default is 80 for HTTP)</p> <p>regexp - regular expression describing the required pattern</p> <p>length - maximum number of characters to return</p> <p>output - an optional output formatting template. The <code>\0</code> escape sequence is replaced with the matched part of text (from the first character where match begins until the character where match</p>	<p>This item turns unsupported if the resource specified path, host does not exist or is unavailable.</p> <p>host can be hostname, domain name, IPv4 or IPv6 address. But for IPv6 address Zabbix agent must be compiled with IPv6 support enabled.</p> <p>Content extraction using the output parameter takes place on the agent.</p> <p>The output parameter is supported from version 2.2.</p> <p>Example: => <code>web.page.regexp[www.examp</code> => <code>web.page.regexp[https://www</code></p>
----------------------------	--	---	---

Chaves de item específicas do Agent 2

O Zabbix Agent 2 suporta todas as chaves de item suportadas pelo Zabbix Agent no **Unix** e no **Windows**. Esta página fornece detalhes sobre as chaves adicionais, as quais você pode usar apenas com Zabbix Agent 2, agrupadas pelo plugin ao qual pertencem.

Veja também: [Plugins fornecidos nativamente](#)

Note:

Parâmetros sem colchetes angulares são obrigatórios. Parâmetros marcados com colchetes angulares < > são opcionais.

Ceph

Chave	Descrição	Valore de retorno	Parâmetros	Comentários
ceph.df.details [connString, <user>, <apikey>]	Uso de dados do cluster e distribuição entre pools.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.	
ceph.osd.stats [connString, <user>, <apikey>]	Estatísticas agregadas e por OSD.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.	
ceph.osd.discovery [connString, <user>, <apikey>]	Lista de OSDs descobertos. Usado para descoberta de baixo-nível .	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.	
ceph.osd.dump [connString, <user>, <apikey>]	Limites de uso e estados de OSDs.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.	
ceph.ping [connString, <user>, <apikey>]				

Chave			
	Testa se uma conexão com o Ceph pode ser estabelecida.	0 - conexão está quebrada (se houver qualquer erro apresentado, incluindo problemas de AUTH e configuração) 1 - conexão bem-sucedida.	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.
ceph.pool.discovery [connString, <user>, <apikey>]	Lista de pools descobertos. Usado para descoberta de baixo-nível.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.
ceph.status [connString, <user>, <apikey>]	Estado geral do cluster.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Ceph.

Docker

Chave				
	Descrição	Valore de retorno	Parâmetros	Comentários
docker.container_info [<ID>]	Informação de baixo-nível sobre um contêiner.	Uma saída da chamada de API ContainerInspect serializada como JSON	ID - ID ou nome do contêiner	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.container_stats [<ID>]	Estatísticas de uso de recursos de contêiner.	Uma saída da chamada de API ContainerStats e porcentagem de uso de CPU serializado como JSON	ID - ID ou nome do contêiner	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.containers				

Chave				
	A list of containers.	Uma saída da chamada de API ContainerList serializada como JSON	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.containers.discovery[<options>]	Uma lista de contêiners. Usado para descoberta de baixo-nível .	Objeto JSON.	options - informa se todos ou apenas os contêiners em execução devem ser descobertos. Valores suportados: true - retorna todos os contêiners; false - retorna apenas os contêiners em execução (padrão).	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.data_usage	Informação sobre o uso de dados atual.	Uma saída da chamada de API System-DataUsage serializada como JSON	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.images	Uma lista de imagens.	Uma saída da chamada de API ImageList serializada como JSON	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.images.discovery				

Chave				
	Uma lista de imagens. Usado para descoberta de baixo-nível .	Objeto JSON.	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.info	Informação de sistema.	Uma saída da chamada de API SystemInfo serializada como JSON	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.
docker.ping	Testa se um daemon de Docker está em execução ou não.	1 - conexão está ativa 0 - conexão está quebrada	-	O usuário do Agent2 ('zabbix') deve ser adicionado ao grupo 'docker' para ter privilégios suficientes. Caso contrário, a verificação irá falhar.

Ember+

Key

Description	Return value	Parameters	Comments
ember.get[<uri>,<path>] Returns the result of the required device.	JSON object	uri - Ember+ device URI. Default: 127.0.0.1:9998 path - OID path to device. Empty by default, returns root collection data.	

Memcached

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
memcached.ping[connString,<user>,<password>]				

Chave				
	Testa se uma conexão está ativa ou não.	1 - conexão está ativa 0 - conexão está quebrada (se houver qualquer erro apresentado, incluindo problemas de AUTH e configuração)	connString - URI ou nome de sessão.	
memcached.stats[connString,<user>,<password>,<type>]	Obtém a saída do comando STATS.	JSON - saída é serializada como JSON	connString - URI ou nome de sessão. user , password - credenciais de login no Memcached. type - tipo de stat a ser retornado: items, sizes, slabs or settings (itens, tamanhos, slabs ou configurações, respectivamente) (vazio por padrão, retorna estatísticas gerais).	

MongoDB

Chave				
	Descrição	Valore de retorno	Parâmetros	Comentários
mongodb.collection.stats[connString,<user>,<password>,<database>,collection]	Retorna uma variedade de estatísticas de armazenamento para uma dada coleção.	Objeto JSON	connString - URI ou nome de sessão. user , password - credenciais de login no MongoDB. database - nome do banco de dados (padrão: admin). collection - nome da coleção.	
mongodb.collections.discovery[connString,<user>,<password>]				

	Retorna uma lista de coleções descobertas. Usado para descobertas de baixo-nível .	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.collections.usage[connString,<user>,<password>]	Retorna estatísticas de uso para coleções.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.connpool.stats[connString,<user>,<password>]	Retorna informação a respeito de conexões de saída abertas da instância de banco de dados atual para outros membros do cluster fragmentado ou conjunto de réplicas.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.db.stats[connString,<user>,<password>,<database>]	Retorna estatísticas refletindo o estado de um dado sistema de banco de dados.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB. database - nome do banco de dados (padrão: admin).
mongodb.db.discovery[connString,<user>,<password>]	Retorna uma lista de banco de dados descobertos. Usado para descobertas de baixo-nível .	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.jumbo_chunks.count[connString,<user>,<password>]			

	Retorna a contagem de jumbo chunks.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.oplog.stats[connString,<user>,<password>]	Retorna o estado do conjunto de réplicas, usando dados consultados do oplog.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.ping[connString,<user>,<password>]	Testa se uma conexão está ativa ou não.	1 - conexão está ativa 0 - conexão está quebrada (se houver qualquer erro presente, incluindo problemas de AUTH e configuração).	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.rs.config[connString,<user>,<password>]	Retorna uma configuração atual do conjunto de réplicas.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.rs.status[connString,<user>,<password>]	Retorna estado de conjunto de réplicas do ponto de vista do membro onde o método é executado.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.server.status[connString,<user>,<password>]	Retorna estado do banco de dados.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.
mongodb.sh.discovery[connString,<user>,<password>]			

Chave			
	Retorna uma lista de fragmentos (shards) descobertos presentes no cluster.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no MongoDB.

MQTT

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
mqtt.get[<broker_url>,topic,<username>,<password>]	Inscreve-se em um tópico específico ou tópicos (com coringa) de um intermediário (broker) fornecido e aguarda por publicações.	Dependendo do conteúdo do tópico. Se coringas são usados, retorna conteúdo de tópico como JSON.	broker_url - URL do intermediário (broker) MQTT (se vazio, é usado localhost com porta 1883). topic - tópico MQTT (obrigatório). Coringas (+,#) são suportados. username,password - credenciais de autenticação (se necessário)	O item deve ser configurado como uma verificação ativa (tipo de item 'Zabbix Agent (ativo)'). Certificados de criptografia TLS podem ser usados salvando-os em local padrão (p.e. diretório /etc/ssl/certs/ para Ubuntu). Para TLS, use o esquema tls:// .

MSSQL

Key			
Description	Return value	Parameters	Comments
mssql.availability.group.get[URI,<user>,<password>] Returns availability groups.	JSON object	URI - MSSQL server URI (the only supported schema is sqlserver://). Embedded credentials will be ignored; user, password - username, password to send to protected MSSQL server.	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.custom.query[URI,<user>,<password>,queryName,<args...>]			

Key			
Returns the result of a custom query.	JSON object	URI - MSSQL server URI (the only supported schema is sqlserver://). Embedded credentials will be ignored; user, password - username, password to send to protected MSSQL server; queryName - name of a custom query configured in <code>Plugins.MSSQL.CustomQueriesDir</code> without the .sql extension; args - one or several comma-separated arguments to pass to a query.	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.db.get			
Returns all available MSSQL databases.	JSON object		Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.job.status.get			
Returns the status of jobs.	JSON object		Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.last.backup.get			
Returns the last backup time for all databases.	JSON object		Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.local.db.get			
Returns databases that are participating in an Always On availability group and replica (primary or secondary) and are located on the server that the connection was established to.	JSON object		Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.mirroring.get			
Returns mirroring info.	JSON object		Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.nonlocal.db.get			

Key		
Returns databases that are participating in an Always On availability group and replica (primary or secondary) located on other servers (the database is not local to the SQL Server instance that the connection was established to).	JSON object	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.perfcounter.get Returns the performance counters.	JSON object	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.ping Ping the database. Test if connection is correctly configured.	1 - alive, 0 - not alive	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.quorum.get Returns the quorum info.	JSON object	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.quorum.members.get Returns the quorum members.	JSON object	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.replica.get Returns the replicas.	JSON object	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.
mssql.version Returns the MSSQL version.	String	Supported since Zabbix 6.0.27. For more information see the MSSQL plugin readme.

MySQL

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
mysql.db.discovery[connString, <username>, <password>]				

mysql.db.size[connString, <username>, <password>, dbName]	Lista de bancos de dados MySQL. Usado para descobertas de baixo-nível.	Resultado da consulta SQL "show databases" em formato LLD JSON.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL.
mysql.get_status_variables[connString, <username>, <password>]	Tamanho do banco de dados em bytes.	Resultado da consulta SQL "select coalesce(sum(data_length + index_length), 0) as size from information_schema.tables where table_schema=?" para o banco de dados específico em bytes.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL. dbName - nome do banco de dados.
mysql.ping[connString, <username>, <password>]	Valores de variáveis globais de estado.	Resultado da consulta SQL "show global status" em formato JSON.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL.
mysql.replication.discovery[connString, <username>, <password>]	Testa se uma conexão está ativa ou não.	1 - conexão está ativa 0 - conexão está quebrada (se houver qualquer erro apresentado, incluindo problemas de AUTH e configuração).	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL.
mysql.replication.get_slave_status[connString, <username>, <password>, <masterHost>]	Lista de replicações MySQL. Usado para descobertas de baixo-nível.	Resultado da consulta SQL "show slave status" em formato LLD JSON.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL.

Chave			
	Estado da replicação.	Resultado da consulta SQL "show slave status" em formato JSON.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL. masterHost - nome do host de replicação principal (master).
mysql.version[connString, <username>, <password>]	Versão MySQL.	String com versão da instância MySQL.	connString - URI ou nome de sessão. username, password - credenciais de login no MySQL.

Oracle

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
oracle.diskgroups.stats[connString,<user>,<password>,<service>]	Estatísticas de grupos de disco ASM.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.	
oracle.diskgroups.discovery[connString,<user>,<password>,<service>]	Lista de grupos de disco ASM. Usado para descobertas de baixo-nível.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.	
oracle.archive.info[connString,<user>,<password>,<service>]				

	Estatísticas de arquivamento de logs.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.cdb.info[connString,<user>,<password>,<service>]	Informação de CDBs.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.custom.query[connString,<user>,<password>,<service>, queryName, <args...>]	Resultado de uma consulta customizada.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle. queryName — nome de uma consulta customizada (deve ser igual ao nome de uma arquivo sql sem uma extensão). args... — um ou vários argumentos separados por vírgula a serem passados em uma consulta.
oracle.datafiles.stats[connString,<user>,<password>,<service>]			

	Estatísticas de arquivos de dados.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.db.discovery[connString,<user>,<password>,<service>]	Lista de bancos de dados. Usado para descobertas de baixo-nível.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.fra.stats[connString,<user>,<password>,<service>]	Estatísticas FRA.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.instance.info[connString,<user>,<password>,<service>]	Estatísticas de instância.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.pdb.info[connString,<user>,<password>,<service>]	Informação sobre PDBs.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.pdb.discovery[connString,<user>,<password>,<service>]			

	Lista de PDBs. Usado para descobertas de baixo-nível.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.pga.stats[connString,<user>,<password>,<service>]	Estatísticas PGA.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.ping[connString,<user>,<password>,<service>]	Testa se uma conexão ao Oracle pode ser estabelecida.	0 - conexão está quebrada (se houver qualquer erro apresentado, incluindo problemas de AUTH e configuração) 1 - conexão bem-sucedida.	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.proc.stats[connString,<user>,<password>,<service>]	Estatísticas de processos.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.redolog.info[connString,<user>,<password>,<service>]	Informação de arquivo de log a partir do arquivo de controle.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.sga.stats[connString,<user>,<password>,<service>]			

	Estatísticas SGA.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.sessions.stats[connString,<user>,<password>,<service>,<lockMaxTime>]	Estatísticas de sessões.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle. lockMaxTime - duração máxima de travamento de sessão em segundos para contar a sessão como prolongada-mente travada. Padrão: 600 segundos.
oracle.sys.metrics[connString,<user>,<password>,<service>,<duration>]	Um conjunto de valores de métricas de sistema.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle. duration - intervalo de captura (em segundos) de valores de métrica de sistema. Valores possíveis: 60 — longa duração (padrão), 15 — curta duração.
oracle.sys.params[connString,<user>,<password>,<service>]			

	Um conjunto de valores de parâmetros de sistema.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.ts.stats[connString,<user>,<password>,<service>]	Estatísticas de tablespaces.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.ts.discovery[connString,<user>,<password>,<service>]	Lista de tablespaces. Usado para descobertas de baixo-nível.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. service - nome de serviço Oracle.
oracle.user.info[connString,<user>,<password>,<username>]	Informação de usuário.	Objeto JSON	connString - URI ou nome de sessão. user, password - credenciais de login no Oracle. username - um nome de usuário, para o qual a informação é necessária. Nomes de usuário em minúsculo não são suportados. Padrão: usuário atual.

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
pgsql.autovacuum.count[uri,<username>,<password>,<dbName>]	O número de processos de trabalho (workers) de autovacuum.	Consulta SQL.	uri - URI ou nome de sessão. username , password - credenciais PostgreSQL. dbName - nome do banco de dados.	
pgsql.archive[uri,<username>,<password>,<dbName>]	Informação sobre arquivos arquivados.	Consulta SQL em formato JSON.	uri - URI ou nome de sessão. username , password - credenciais PostgreSQL. dbName - nome do banco de dados.	Os dados retornados são processados por itens dependentes: pgsql.archive.count_archived - o número de arquivos WAL que foram arquivados com sucesso. pgsql.archive.failed_trying - o número de tentativas falhas de arquivamento de arquivos WAL. pgsql.archive.count_files_to_archive - o número de arquivos a arquivar. pgsql.archive.size_files_to_archive - o tamanho dos arquivos a arquivar.
pgsql.bgwriter[uri,<username>,<password>,<dbName>]				

Número combinado de pontos de verificação (checkpoints) para o cluster de banco de dados, dividido por tipo de checkpoint.

Consulta SQL em formato JSON.

uri - URI ou nome de sessão.
username,
password - credenciais PostgreSQL.
dbName - nome do banco de dados.

Os dados retornados são processados por itens dependentes:
pgsql.bgwriter.buffers_allot
 - o número de buffers alocados.
pgsql.bgwriter.buffers_bac
 - o número de buffers escritos diretamente por um backend.
pgsql.bgwriter.maxwritten
 - o número de vezes que um processo de background writer parou um scan de limpeza, porque havia escrito buffers demais.
pgsql.bgwriter.buffers_bac
 - o número de vezes que um backend teve que executar sua própria chamada de fsync em vez do processo de background writer.
pgsql.bgwriter.buffers_clea
 - o número de buffers escritos pelo processo de background writer.
pgsql.bgwriter.buffers_che
 - o número de buffers escritos durante checkpoints.
pgsql.bgwriter.checkpoints
 - o número de checkpoints agendados que foram executados.
pgsql.bgwriter.checkpoints
 - o número de checkpoints requisitados que foram executados.
pgsql.bgwriter.checkpoint_
 - a quantidade total de tempo gasto na porção de

Chave

pgsql.cache.hit[uri,<username>,<password>,<dbName>]

Taxa de acerto
do cache de
buffer
PostgreSQL.

Consulta SQL
em
porcentagem.

uri - URI ou
nome de
sessão.
username,
password -
credenciais
PostgreSQL.
dbName -
nome do banco
de dados.

pgsql.connections[uri,<username>,<password>,<dbName>]

Conexões por tipo.

Objeto JSON.

uri - URI ou nome de sessão.
username, password - credenciais PostgreSQL.
dbName - nome do banco de dados.

Os dados retornados são processados por itens dependentes:

pgsql.connections.active
 - o backend está executando uma consulta.

pgsql.connections.fastpath
 - o backend está executando uma função de caminho-rápido (fast-path).

pgsql.connections.idle
 - o backend está aguardando por um novo comando de cliente.

pgsql.connections.idle_in_t
 - o backend está em uma transação, mas não está executando uma consulta no momento.

pgsql.connections.prepare
 - o número de conexões preparadas.

pgsql.connections.total
 - o número total de conexões.

pgsql.connections.total_pc
 - porcentagem do total de conexões em relação à configuração 'max_connections' do servidor PostgreSQL.

pgsql.connections.waiting
 - número de conexões em uma consulta.

pgsql.connections.idle_in_t
 - o backend está em uma transação, mas não está executando uma consulta no momento e uma das declarações na consulta causou um

Chave

pgsql.dbstat[uri,<username>,<password>,
dbName]

Coleta estatísticas por banco de dados. Usado para descobertas de baixo-nível.

Consulta SQL em formato JSON.

uri - URI ou nome de sessão.
username,
password - credenciais PostgreSQL.
dbName - nome do banco de dados.

Os dados retornados são processados por itens dependentes:
pgsql.dbstat.numbackends
 - tempo gasto lendo blocos de arquivo de dados por backends neste banco de dados, em milissegundos.
pgsql.dbstat.sum.blk_read
 - tempo gasto lendo blocos de arquivo de dados por backends neste banco de dados, em milissegundos.
pgsql.dbstat.sum.blk_write
 - tempo gasto escrevendo blocos de arquivo de dados por backends neste banco de dados, em milissegundos.
pgsql.dbstat.sum.checksum
 - o número de falhas de checksum de páginas de dados detectadas (ou um objeto compartilhado), ou NULL se checksums de dados não estão habilitados. (apenas PostgreSQL versão 12)
pgsql.dbstat.blks_read.rate
 - o número de leitura de blocos de disco neste banco de dados.
pgsql.dbstat.deadlocks.rate
 - o número de deadlocks detectados neste banco de dados.
pgsql.dbstat.blks_hit.rate
 - o número de vezes em que blocos de disco

Chave

pgsql.dbstat.sum[uri,<username>,<password>,
<dbName>]

Dados resumidos para todos os bancos de dados em um cluster.

Consulta SQL em formato JSON.

uri - URI ou nome de sessão.
username,
password - credenciais PostgreSQL.
dbName - nome do banco de dados.

Os dados retornados são processados por itens dependentes:
pgsql.dbstat.numbackends
 - o número de backends atualmente conectados a este banco de dados.
pgsql.dbstat.sum.blk_read
 - tempo gasto lendo blocos de arquivo de dados por backends neste banco de dados, em milissegundos.
pgsql.dbstat.sum.blk_write
 - tempo gasto escrevendo blocos de arquivos de dados por backends neste banco de dados, em milissegundos.
pgsql.dbstat.sum.checksum
 - o número de falhas de checksum de página de dados detectado (ou em um objeto compartilhado), ou NULL se checksums de dados não estiverem habilitados (apenas PostgreSQL versão 12).
pgsql.dbstat.sum.xact_com
 - o número de transações neste banco de dados que foram entregues.
pgsql.dbstat.sum.conflicts
 - estatísticas de banco de dados sobre consultas canceladas devido conflito com recuperação em servidores

Chave

pgsql.db.age[uri,<username>,<password>,
dbName]

Idade do
FrozenXID mais
antigo do
banco de
dados. Usado
para
**descobertas de
baixo-nível.**

Consulta SQL
para banco de
dados
específico em
transações.

uri - URI ou
nome de
sessão.
username,
password -
credenciais
PostgreSQL.
dbName -
nome do banco
de dados.

pgsql.db.bloating_tables[uri,<username>,<password>,
<dbName>]

O número de
bloating tables
por banco de
dados. Usado
para
**descobertas de
baixo-nível.**

Consulta SQL.

uri - URI ou
nome de
sessão.
username,
password -
credenciais
PostgreSQL.
dbName -
nome do banco
de dados.

pgsql.db.discovery[uri,<username>,<password>,
<dbName>]

Lista de bancos
de dados
PostgreSQL.
Usado para
**descobertas de
baixo-nível.**

Consulta SQL
no formato LLD
JSON.

uri - URI ou
nome de
sessão.
username,
password -
credenciais
PostgreSQL.
dbName -
nome do banco
de dados.

pgsql.db.size[uri,<username>,<password>,
dbName]

Tamanho do
banco de
dados em
bytes. Usado
para
**descobertas de
baixo-nível.**

Consulta SQL
para banco de
dados
específico em
bytes.

uri - URI ou
nome de
sessão.
username,
password -
credenciais
PostgreSQL.
dbName -
nome do banco
de dados.

pgsql.locks[uri,<username>,<password>,
<dbName>]

Chave				
	<p>Informação sobre bloqueios de permissão (granted locks) por banco de dados. Usado para descobertas de baixo-nível.</p>	<p>Consulta SQL em formato JSON.</p>	<p>uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.</p>	<p>Os dados retornados são processados por itens dependentes:</p> <p>pgsql.locks.shareupdateexclusive - o número de bloqueios Share Update Exclusive.</p> <p>pgsql.locks.accessexclusive - o número de bloqueios Access Exclusive.</p> <p>pgsql.locks.accessshare - o número de bloqueios Access Share.</p> <p>pgsql.locks.exclusive - o número de bloqueios Exclusive.</p> <p>pgsql.locks.rowexclusive - o número de bloqueios Row Exclusive.</p> <p>pgsql.locks.rowshare - o número de bloqueios Row Share.</p> <p>pgsql.locks.share - o número de bloqueios Shared.</p> <p>pgsql.locks.sharerowexclusive - o número de bloqueios Share Row Exclusive.</p>
<p>pgsql.oldest.xid[uri,<username>,<password>,<dbName>]</p>	<p>Idade do XID mais antigo.</p>	<p>Consulta SQL.</p>	<p>uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.</p>	
<p>pgsql.ping[uri,<username>,<password>,<dbName>]</p>				

Chave

	Testa se uma conexão está ativa ou não.	1 - conexão está ativa 0 - conexão está quebrada (se há qualquer erro apresentado, incluindo problemas de AUTH e configuração).	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.
<code>pgsql.replication.count[uri,<username>,<password>,<dbName>]</code>	O número de servidores standby.	Consulta SQL.	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.
<code>pgsql.replication.recovery_role[uri,<username>,<password>,<dbName>]</code>	Estado de recuperação.	0 - modo master 1 - recuperação ainda está em progresso (modo espera (standby))	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.
<code>pgsql.replication.status[uri,<username>,<password>,<dbName>]</code>	O estado da replicação.	0 - streaming está fora 1 - streaming está no ar 2 - modo master	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.
<code>pgsql.replication_lag.b[uri,<username>,<password>,<dbName>]</code>	Atraso de replicação em bytes.	Consulta SQL em bytes.	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.
<code>pgsql.replication_lag.sec[uri,<username>,<password>,<dbName>]</code>			

Chave				
	Atraso de replicação em segundos.	Consulta SQL em segundos.	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.	
pgsql.uptime[uri,<username>,<password>,<dbName>]	PostgreSQL uptime em milisegundos.	Consulta SQL em milisegundos.	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados.	
pgsql.wal.stat[uri,<username>,<password>,<dbName>]	Estatísticas de WAL.	Consulta SQL em formato JSON.	uri - URI ou nome de sessão. username, password - credenciais PostgreSQL. dbName - nome do banco de dados. Os dados retornados são processados por itens dependentes: pgsql.wal.count - o número de arquivos WAL. pgsql.wal.write - o WAL Isn usado (em bytes).	

Redis

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
redis.config[connString,<password>,<pattern>]	Obtém os parâmetros de configuração de uma instância de Redis que corresponde ao padrão.	JSON - se um padrão glob-style for usado valor único - se o padrão não contém qualquer caracter coringa	connString - URI ou nome de sessão. password - senha Redis. pattern - padrão glob-style (* por padrão).	
redis.info[connString,<password>,<section>]				

Chave			
	Obtém a saída do comando INFO.	JSON - saída é serializada como JSON	connString - URI ou nome de sessão. password - senha Redis. section - seção de informação (por padrão configurado como padrão).
redis.ping[connString,<password>]	Testa se uma conexão está ativa ou não.	1 - conexão está ativa 0 - conexão está quebrada (se houver qualquer erro apresentado, incluindo problemas de AUTH e configuração)	connString - URI ou nome de sessão. password - senha Redis.
redis.slowlog.count[connString,<password>]	O número de entradas de log lentas desde que o Redis foi iniciado.	Inteiro	connString - URI ou nome de sessão. password - senha Redis.

S.M.A.R.T.

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
smart.attribute.discovery				

Chave			
	Retorna uma lista de atributos de dispositivo S.M.A.R.T.	Objeto JSON	As seguintes macros e seus valores são retornados: {#NAME}, {#DISKTYPE}, {#ID}, {#ATTRNAME}, {#THRESH}. Os tipos de drive HDD, SSD e NVME são suportados. Os drives podem estar sozinhos ou combinados em um RAID. {#NAME} terá um add-on em caso de RAID, p.e.: {"{#NAME}": "/dev/sda cciss,2"}
smart.disk.discovery	Retorna uma lista de dispositivos S.M.A.R.T.	Objeto JSON	As seguintes macros e seus valores são retornados: {#NAME}, {#DISKTYPE}, {#MODEL}, {#SN}. Os tipos de drive HDD, SSD e NVME são suportados. Os drives podem estar sozinhos ou combinados em um RAID. {#NAME} terá um add-on em caso de RAID, p.e.: {"{#NAME}": "/dev/sda cciss,2"}
smart.disk.get			

Chave			
	Retorna todas as propriedades disponíveis de dispositivos S.M.A.R.T.	Objeto JSON	Os tipos de drive HDD, SSD e NVME são suportados. Os drives podem estar sozinhos ou combinados em um RAID. Os dados incluem versão smartctl e chamada de argumentos, e campos adicionais: disk_name - contém o nome com os add-ons necessários para descoberta de RAID, p.e.: {"disk_name": "/dev/sda cciss,2"} disk_type - contém o nome do tipo de disco HDD, SSD, ou NVME, p.e.: {"disk_type": "ssd"} }

Systemd

Chave				
	Descrição	Valor de retorno	Parâmetros	Comentários
systemd.unit.get[unit name,<interface>]				

Chave

	Retorna todas as propriedades de uma unidade systemd.	Objeto JSON	unit name - nome da unidade (você pode querer usar a macro {#UNIT.NAME} no protótipo de item para descobrir o nome) interface - tipo de interface da unidade, valores possíveis: Unit (padrão), Service, Socket, Device, Mount, Automount, Swap, Target, Path	Este item é suportado apenas na plataforma Linux. LoadState, ActiveState e UnitFileState para a interface Unit são retornados como texto e inteiro: "ActiveState":{"state":
systemd.unit.info[unit name,<property>,<interface>]				

Chave	Informação sobre unidade (unit) Systemd.	String	unit name - nome da unidade (você pode querer usar a macro {#UNIT.NAME} no protótipo de item para descobrir o nome) property - propriedade da unidade (p.e. ActiveState (padrão), LoadState, Description) interface - tipo de interface da unidade (p.e. Unit (padrão), Socket, Service)	Este item permite obter uma propriedade específica de um tipo de interface específico, como descrito em API dbus . Este item é suportado apenas na plataforma Linux. Exemplos: => sys-temd.unit.info["{#UNIT.NAME} - coleta informação de estado ativo (active, reloading, inactive, failed, activating, deactivating (ativo, carregando, inativo, falhou, ativando e desativando, respectivamente)) sobre unidades do systemd descobertas => sys-temd.unit.info["{#UNIT.NAME} - coleta informação de estado de carregamento sobre unidades do systemd descobertas => sys-temd.unit.info[mysqld.service - obtém nome técnico do serviço (mysqld.service) => sys-temd.unit.info[mysqld.service - obtém descrição do serviço (MySQL Server) => sys-temd.unit.info[mysqld.service - obtém a última data em que o serviço entrou
-------	--	--------	--	---

Chave

systemd.unit.discovery[<type>]

Lista de unidades de systemd e seus detalhes.
Usado para descoberta de baixo-nível.

Objeto JSON

type - valores possíveis: all, automount, device, mount, path, service (padrão), socket, swap, target

Este item é suportado apenas na plataforma Linux.

Certificado web

Chave

	Descrição	Valore de retorno	Parâmetros	Comentários
web.certificate.get[hostname,<port>,<address>]				

Valida certificados e retorna detalhes de certificado.	Objeto JSON	<p>hostname - pode ser IP ou DNS. Pode conter o esquema de URL (https apenas), caminho (ele será ignorado), e porta. Se uma porta é fornecida em ambos primeiro e segundo parâmetros, seus valores devem coincidir. Se um endereço (o 3º parâmetro) é especificado, o nome de host é usado apenas para SNI e verificação de nome de host.</p> <p>port - número de porta (padrão é 443 para HTTPS).</p> <p>address - pode ser IP ou DNS. Se informado, ele será usado para conexão, e o nome de host (o 1º parâmetro) será usado para SNI, e verificação de host.</p> <p>No caso de o 1º parâmetro ser um IP e o 3º parâmetro um DNS, o 1º parâmetro será usado para conexão, e o 3º parâmetro será usado para SNI e verificação de nome de host.</p>	<p>Este item se torna não suportado se o recurso especificado em <code>host</code> não existir ou estiver indisponível ou se a negociação TLS falhar com algum erro diferente de certificado inválido.</p> <p>Atualmente, a extensão AIA (Authority Information Access) X.509, CRLs e OCSP (incluindo grampeamento OCSP), Transparência de Certificado, e CA Trust Store personalizados não são suportados.</p>
--	-------------	---	---

Chaves de item

A tabela oferece detalhes das chaves de item que você pode usar apenas com agente Windows.

Veja também: [Nível mínimo de permissão para itens de agente Windows](#)

Chave	Descrição	Valor de retorno	Parâmetros	Comentários
eventlog	[name,<regexp>,<severity>,<source>,<eventid>,<maxlines>,<mode>]			

Monitoramento de log de evento.	Log	<p>name - nome do log de evento</p> <p>regex - expressão regular descrevendo o padrão desejado</p> <p>severity - expressão regular descrevendo a severidade</p> <p>Este parâmetro aceita os seguintes valores:</p> <p>"Information", "Warning", "Error", "Critical", "Verbose"</p> <p>(Informação, Alerta, Erro, Crítico e Detalhado, respectivamente) (desde o Zabbix 2.2.0, executando em Windows Vista ou mais recente)</p> <p>source - expressão regular descrevendo o identificador de origem (expressão regular é suportada desde o Zabbix 2.2.0)</p> <p>eventid - expressão regular descrevendo o(s) identificador(es) de evento</p> <p>maxlines - número máximo de novas linhas por segundo que o agente enviará para o Zabbix Server ou Proxy. Este parâmetro sobrescreve o valor de 'MaxLinesPer-</p>	<p>O item deve ser configurado como uma verificação ativa.</p> <p>Exemplos:</p> <p>=> event-log[Application]</p> <p>=> event-log[Security,"Failure Audit","^(529 680)\$]</p> <p>=> event-log[System,"Warning Error"]</p> <p>=> event-log[System,,,,^1\$]</p> <p>=> event-log[System,,,,@TWOSHORT]</p> <p>- aqui uma expressão regular customizada com nome TWOSHORT é referenciada (definida como do tipo Resultado é VERDADEIRO, sendo a própria expressão ^1\$\ ^70\$).</p> <p>Note que o agente é incapaz de enviar eventos a partir do log de "Eventos encaminhados" (Forwarded events).</p> <p>O parâmetro mode é suportado desde o Zabbix 2.0.0. "Windows Eventing 6.0" é suportado desde o Zabbix 2.2.0.</p> <p>Note que selecionando um tipo de informação não-log para este item levará à perda da identificação de data local</p>
---------------------------------	-----	---	---

Chave

net.if.list

Lista de interface de rede (inclui tipo de interface, estado, endereço IPv4, descrição).

Texto

Suportado desde a versão de Zabbix Agent 1.8.1. Nomes de interface multi-byte suportados desde a versão de Zabbix Agent 1.8.6. Interfaces desabilitadas não são listadas.

Note que habilitar/desabilitar alguns componentes pode alterar sua ordenação no nome de interface do Windows.

Algumas versões de Windows (por exemplo, Server 2008) podem exigir as últimas atualizações instaladas de modo a suportar caracteres não-ASCII nos nomes de interface.

perf_counter[counter,<interval>]

Valor de qualquer contador de performance do Windows.

Inteiro, número flutuante, string ou texto (dependendo da requisição)

counter - caminho para o contador
interval - últimos N segundos para armazenar o valor médio. O valor (**interval**) deve estar entre 1 e 900 segundos (este inclusive) e o valor padrão é 1.

Monitor de Performance pode ser usado para obter uma lista de contadores disponíveis. Até a versão 1.6 este parâmetro retornará valores corretos apenas para contadores que requerem uma única amostra (como \System\Threads). Ele não trabalhará como esperado com contadores que requerem mais de uma amostra - como utilização de CPU. Desde 1.6, **interval** é usado, então a verificação retorna um valor médio dos últimos "intervalo" segundos toda vez.

Veja também:
[Contadores de performance do Windows.](#)

`perf_counter_en[counter,<interval>]`

Chave				
	Valor de qualquer contador de performance do Windows em Inglês.	Inteiro, número flutuante, string ou texto (dependendo da requisição)	counter - caminho para o contador em Inglês interval - últimos N segundos para armazenar o valor médio. O valor (interval) deve estar entre 1 e 900 segundos (este inclusive) e o valor padrão é 1.	Este item é suportado apenas em Windows Server 2008/Vista e superior. Você pode encontrar a lista de strings em Inglês pela visualização da seguinte chave de registro: <code>HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Performance</code>
perf_instance.discovery[object]	Lista de instâncias de objeto de contadores de performance do Windows. Usado para descoberta de baixo-nível .	Objeto JSON	object - nome do objeto (localizado)	Suportado desde as versões de Zabbix Agent 4.0.13 e 4.2.7. Suportado desde a versão de Zabbix Agent 5.0.1.
perf_instance_en.discovery[object]	Lista de instâncias de objeto de contadores de performance do Windows, descobertos usando nomes de objeto em Inglês. Usado para descoberta de baixo-nível .	Objeto JSON	object - nome do objeto (em Inglês)	Suportado desde a versão de Zabbix Agent 5.0.1.
proc_info[process,<attribute>,<type>]				

Várias
informações
sobre
processo(s)
específico(s).

Número
flutuante

process -
nome do
processo
attribute -
atributo do
processo
requerido
type - tipo de
representação
(significante
quando mais
de um
processo com
mesmo nome
existir)

Os seguintes
atributos
(attribute)
são
suportados:
vmsize
(padrão) -
tamanho da
memória
virtual do
processo em
Kbytes
wkset -
tamanho do
conjunto de
operação do
processo
(quantidade de
memória física
usada por
processo) em
Kbytes
pf - número de
faltas de
página
ktime - tempo
de kernel do
processo em
milissegundos
utime - tempo
de usuário do
processo em
milissegundos
io_read_b -
número de
bytes lidos por
processo
durante
operações de
I/O
io_read_op -
número de
operações de
leitura
executadas por
processo
io_write_b -
número de
bytes escritos
por processo
durante
operações de
I/O
io_write_op -
número de
operações de
escrita
executadas por
processo
io_other_b -
número de
bytes
transferidos
por processo
durante outras

Chave			
service.discovery	Lista de serviços Windows. Usado para descoberta de baixo-nível.	Objeto JSON	Suportado desde a versão de Zabbix Agent 3.0.
service.info[service,<param>]			

Informação sobre um serviço.	Inteiro - com param como state, startup (estado, inicialização)	service - um nome de serviço real ou seu nome de exibição como visto na tela de Serviços MMC	Exemplos: => ser-vice.info[SNMPTRAP] - estado do serviço SNMPTRAP => ser-vice.info[SNMPTrap] - estado do mesmo serviço, mas com nome de exibição especificado => ser-vice.info[EventLog,startup] - tipo de inicialização do serviço de log de eventos (EventLog)
	String - com param como displayname, path, user (nome de exibição, caminho, usuário)	param - state (padrão), displayname, path, user, startup ou description (estado, nome de exibição, caminho, usuário, inicialização ou descrição, respectivamente)	Os itens ser-vice.info[service,state] e ser-vice.info[service] retornarão a mesma informação.
	Texto - com param como description (descrição)		Note que apenas com param como state este parâmetro retorna um valor para serviços não existentes (255).
	Especificamente para state (estado): 0 - em execução, 1 - pausado, 2 - execução pendente, 3 - pausa pendente, 4 - continuar pendente, 5 - parada pendente, 6 - parado, 7 - desconhecido, 255 - serviço não existente		
	Especificamente para startup (inicialização): 0 - automática, 1 - automática com atraso, 2 - manual, 3 - desabilitada, 4 - desconhecida, 5 - início com gatilho automático, 6 - início com atraso e gatilho automático, 7 - início com gatilho manual		Este item é suportado desde o Zabbix 3.0.0. Ele deve ser usado em vez do item descontinuado ser-vice_state[service].

services[<type>,<state>,<exclude>]

Listagem de serviços.	0 - se vazio Texto - lista de serviços separados por uma nova linha	type - all (padrão), automatic, manual ou disabled (tudo, automático, manual ou desabilitado, respectivamente) state - all (padrão), stopped, started, start_pending, stop_pending, running, continue_pending, pause_pending ou paused (tudo, parado, iniciado, início pendente, parada pendente, em execução, continuar pendente, pausa pendente ou pausado, respectivamente) exclude - serviços a excluir do resultado. Serviços excluídos devem ser listados entre aspas duplas, separados por vírgula, sem espaços.	Exemplos: => services[,started] - lista de serviços iniciados => services[automatic, stopped] - lista de serviços parados, que devem ser executados => services[automatic, stopped, "service1,service2,service3"] - lista de serviços parados, que devem ser executados, excluindo os serviços com os nomes service1, service2 e service3 O parâmetro exclude é suportado desde o Zabbix 1.8.1.
-----------------------	--	--	--

wmi.get[<namespace>,<query>]

Chave				
	Executa uma consulta WMI e retorna o primeiro objeto selecionado.	Inteiro, número flutuante, string ou texto (dependendo da requisição)	namespace - namespace WMI query - consulta WMI retornando um objeto único	Consultas WMI são realizadas com WQL . Exemplo: => wmi.get[root\cimv2,select status from Win32_DiskDrive where Name like '%PHYSICALDRIVE0%'] - retorna o estado do primeiro disco físico Esta chave é suportada desde o Zabbix 2.2.0.
wmi.getall[<namespace>,<query>]	Executa consulta WMI e retorna resposta completa. Pode ser usada para descoberta de baixo-nível .	Objeto JSON	namespace - namespace WMI query - consulta WMI	Consultas WMI são realizadas com WQL . Exemplo: => wmi.getall[root\cimv2,select * from Win32_DiskDrive where Name like '%PHYSICALDRIVE%'] - retorna informação de estado dos discos físicos Pré-processamento com JSONPath pode ser usado para apontar para valores mais específicos no JSON retornado. Esta chave é suportada desde o Zabbix 4.4.0.
vm.memory.size[<type>]				

Chave				
	Tamanho de memória virtual em bytes ou em porcentagem do total.	Inteiro - para bytes Número flutuante - para porcentagem	type - valores possíveis: available (memória virtual disponível), pavailable (memória virtual disponível, em por cento), pused (memória virtual usada, em por cento), total (memória virtual total, padrão), used (memória virtual usada)	Exemplo: => vm.vmemory.size[pavailable] → memória virtual disponível, em porcentagem O monitoramento de estatísticas de memória virtual é baseado em: * Memória virtual total no Windows (física total + tamanho arquivo de página); * A quantidade máxima de memória que o Zabbix Agent pode comprometer; * O limite atual de memória comprometida para o sistema ou Zabbix Agent, o qual for menor. Esta chave é suportada desde o Zabbix 3.0.7 e 3.2.3.

Monitorando serviços do Windows

Este tutorial provê um passo a passo de como se configurar a monitoramento de serviços do Windows. Partimos do princípio de que o Zabbix Agent e o Zabbix Server estão configurados e operacionais.

Passo 1

Obter o nome do serviço.

Você pode obter o nome do serviço ao acessar a tela de Serviços no MMC e analisar as propriedades do serviço desejado. Na aba "Geral" você deve procurar pelo campo 'Nome do serviço'. Você usará o valor deste campo para configurar um item para monitoramento.

Por exemplo, se você quiser monitorar o serviço "Estação de trabalho" então o nome do serviço seria: **LanmanWorkstation**.

Passo 2

Configure um item para monitorar o serviço.

O item 'service.info[service,<param>]' obtém informações sobre um serviço em particular. Dependendo da informação que você precisar, defina a opção param para um destes valores: displayname, state, path, user, startup ou description (nome de exibição, estado, caminho, usuário, inicialização ou descrição, respectivamente). O valor padrão é state se param não for definido (service.info[service]).

O tipo do valor de retorno depende do que for definido no parâmetro param: inteiro para state e startup; string para displayname, path e user; texto para description.

Exemplo:

- Chave: service.info[lanmanworkstation]
- Tipo de informação: Numérico (unsigned)
- Mostrar valor: selecione o mapeamento de valor para estado de serviço Windows

Dois mapeamentos de valor estão disponíveis, "Estado de serviço Windows" e "Tipo de inicialização de serviço Windows", para mapear um valor numérico obtido para uma representação textual na interface web do Zabbix.

Descoberta de serviços do Windows

A **descoberta de baixo-nível** provê um caminho para a criação automática de itens, gatilhos, e gráficos para diferentes entidades em um computador. O Zabbix pode monitorar automaticamente os serviços Windows de sua máquina, sem que você precise saber exatamente o nome do serviço para criar os itens manualmente. Um filtro pode ser utilizado para criar de fato itens, gatilhos, e gráficos apenas para os serviços de interesse.

2 Agente SNMP

Visão geral

Você poderá precisar utilizar o protocolo SNMP para monitorar dispositivos como impressoras, switches, roteadores ou nobreaks que, normalmente, possuem interfaces SNMP habilitadas e onde é impraticável manter um Zabbix Agent funcionando.

Para que o Zabbix Server esteja apto a receber dados coletados por um agente SNMP, ele deverá ser **configurado** com o suporte SNMP.

As coletas SNMP são feitas somente através do protocolo UDP.

Desde o Zabbix 2.2.3 o Zabbix Server e o Zabbix Proxy tem a capacidade de coletar múltiplos dados de um dispositivo a partir de uma única requisição. Isso afeta todos os tipos de itens SNMP (itens normais SNMP, itens SNMP com índice dinâmico, e o processo de autobusca SNMP) os tornando mais eficientes. Mais detalhes sobre **coletas múltiplas** podem ser obtidos. Desde o Zabbix 2.4 existe uma opção chamada "Usar requisições em lote" em cada interface SNMP que permite habilitar ou desabilitar requisições em lote naquela interface.

Desde o Zabbix 2.2.7 e o Zabbix 2.4.2 os processos do Zabbix Server e do Zabbix Proxy registram de forma similar respostas incorretas do SNMP:SNMP response from host "gateway" does not contain all of the requested variable bindings Enquanto não forem mapeadas todas as situações problemáticas, elas são úteis para identificar dispositivos cuja coleta em lote (bulk) pode ser desabilitada.

Desde o Zabbix 2.2 os processos do Zabbix Server e do Zabbix Proxy utilizam o parâmetro de 'Timeout' nas requisições SNMP. Adicionalmente eles não realizam o reteste após uma requisição que falhar (por 'timeout' ou credenciais erradas). Anteriormente era padrão que a biblioteca SNMP tivesse o 'timeout' de 1 segundo e 5 tentativas.

Desde o Zabbix 2.2.8 e o Zabbix 2.4.2 os processos do Zabbix Server e do Zabbix Proxy sempre irão repetir pelo menos uma vez, seja através do mecanismo interno da biblioteca SNMP ou através do **mecanismo interno de processamento em lote**.

Warning:

Se estiver monitorando dispositivos SNMPv3, certifique-se que o 'msgAuthoritativeEngineID' (também conhecido como 'snmpEngineID' or "Engine ID") nunca seja repetido entre dois dispositivos. Conforme a [RFC 2571](#) (seção 3.1.1.1) este valor deve ser único para cada dispositivo.

Configurando monitoramento SNMP

Para iniciar o monitoramento de um dispositivo através do protocolo SNMP é necessário executar estas etapas:

Passo 1

Criar um host para o dispositivo que possui interface SNMP.

Informe o endereço IP de sua interface. Você pode aproveitar um dos templates de SNMP (Template SNMP Device e outros) que são fornecidos junto com o Zabbix, eles irão adicionar automaticamente um conjunto de itens. Entretanto, o dispositivo que você deseja monitorar poderá não ser compatível com eles. Clique em Adicionar para salvar o host.

Note:

As requisições SNMP não utilizam a porta do agente, ela será ignorada.

Passo 2

Descubra o OID que você deseja monitorar (normalmente na documentação do dispositivo).

Para obter uma lista dos OIDs do dispositivo o comando **snmpwalk** poderá ser utilizado (ele é parte do pacote [net-snmp](#) que você adicionou durante a instalação do Zabbix Server) ou alguma ferramenta equivalente:

```
shell> snmpwalk -v 2c -c public <host IP> .
```

O '2c' utilizado no comando se refere à versão do SNMP, caso seu dispositivo trabalhe, por exemplo, com a primeira versão do protocolo você pode substituí-lo por '1', para indicar que o 'snmpwalk' deverá utilizar esta versão do protocolo.

Você deverá ter como resultado uma lista contendo os nomes dos OIDs, seus tipos e último valor de cada um. É possível que o nome de comunidade configurada no dispositivo seja outro que não o 'public', neste caso você deverá descobrir o nome correto e utilizá-lo.

Você pode recorrer a esta lista para localizar qual nome de OID que você deseja monitorar, por exemplo, se você desejar monitorar o volume em bytes que é recebido pela terceira porta do dispositivo, você poderia utilizar o nome de OID IF-MIB::ifInOctets.3, conforme o exemplo a seguir:

```
IF-MIB::ifInOctets.3 = Counter32: 3409739121
```

Agora utilizaremos o comando **snmpget** para descobrir o número associado ao nome de OID selecionado ('IF-MIB::ifInOctets.3'):

```
shell> snmpget -v 2c -c public -On 10.62.1.22 IF-MIB::ifInOctets.3
```

Observe que o último número neste nome de OID é o número da porta que você deseja monitorar. Mais detalhes em [índices dinâmicos do SNMP](#).

O resultado do comando deve ser algo similar ao texto a seguir:

```
.1.3.6.1.2.1.2.2.1.10.3 = Counter32: 3472126941
```

Novamente, o último número no OID é o número da porta.

Note:

Alguns fabricantes, como a 3COM, preferem nomear as suas portas começando no número 100, exemplos: porta 1 = porta 101, porta 3 = porta 103. A CISCO (e a maioria dos fornecedores) usa uma sequência simples, exemplo: port 3 = 3.

Note:

Alguns dos OIDs mais utilizados são [traduzidos automaticamente para a representação numérica](#) de forma nativa no Zabbix.

No último exemplo o tipo do valor recebido é "Counter32", que corresponde internamente ao tipo ASN_COUNTER. A lista completa de tipos suportados é: ASN_COUNTER, ASN_COUNTER64, ASN_INTEGER, ASN_UNSIGNED64, ASN_INTEGER64, ASN_FLOAT, ASN_DOUBLE, ASN_TIMETICKS, ASN_GAUGE, ASN_IPADDRESS, ASN_OCTET_STR e ASN_OBJECT_ID (desde o Zabbix 2.2.8 ou 2.4.3). Estes tipos correspondem, respectivamente, à: "Counter32", "Counter64", "UInteger32", "INTEGER", "Float", "Double", "Timeticks", "Gauge32", "IpAddress", "OCTET STRING", "OBJECT IDENTIFIER" na saída do **snmpget**, mas também podem ser apresentados como "STRING", "Hex-STRING", "OID" ou outro.

Passo 3

Criar o item de monitoração.

Volte ao Zabbix e clique no link Itens da linha do Host SNMP que você criou mais cedo. Dependendo da associação que você fez, ou não fez, com algum template você poderá visualizar uma lista com ou sem itens. Neste tutorial estamos trabalhando assumindo que você criará manualmente os itens a partir das informações conseguidas pelo 'snmpwalk' e 'snmpget'. Clique em Criar item. No formulário de novo item, informe o nome e se certifique que o campo 'Interface do host' está com a configuração da interface SNMP cadastrada. Informe o nome de comunidade (normalmente 'public') e o número do OID que se deseja monitorar, por exemplo: .1.3.6.1.2.1.2.2.1.10.3

Informe a porta SNMP como sendo a 161 e 'chave' como algo significativo, por exemplo: SNMP-InOctets-Bps. Informe o multiplicador que se deseja trabalhar, além de informar o 'Intervalo de atualizações' e o 'Período de retenção do histórico' se você desejar usar um valor diferente do padrão. Defina o 'Tipo de informação' para Numérico (fracionário) e o campo 'Armazenar valor' para "Delta (alterações por segundo)" (isso é importante, sem esta configuração você terá o acumulado de valores e não é isso que se deseja).

Items

All hosts / Zabbix server Enabled ZBX SNMP JMX IPMI Applications 13 Items 81 Triggers 47

Item Preprocessing

* Name	SNMP: InOctets (Bps)
Type	SNMPv3 agent
* Key	SNMP-InOctets-Bps
* Host interface	127.0.0.1 : 161
* SNMP OID	.1.3.6.1.2.1.2.2.1.10.3
Context name	
Security name	
Security level	authPriv
Authentication protocol	MD5 SHA
Authentication passphrase	
Privacy protocol	DES AES
Privacy passphrase	
Port	161
Type of information	Numeric (float)

Agora salve o item e acesse Monitoramento → Dados recentes para visualizar a coleta de dados!

Observe que existem opções específicas para o SNMPv3:

Parâmetro	Descrição	
Nome do contexto	Informe o nome para identificar o item na subrede SNMP. Suportado desde o Zabbix 2.2. Macros de usuário são aceitas neste campo. Nome de segurança//	Informe o nome de segurança. Macros de usuário são aceitas neste campo.

Parâmetro	Descrição
Nível de segurança	<p>Selecione o nível de segurança:</p> <p>noAuthNoPriv - não será utilizado nenhum protocolo de privacidade ou autenticação</p> <p>AuthNoPriv - protocolo de autenticação será utilizado, de privacidade não</p> <p>AuthPriv - ambos os protocolos serão utilizados</p>
Protocolo de autenticação	Selecione o tipo de protocolo de autenticação - MD5 ou SHA.
Senha autenticação	Informe a frase de autenticação. Macros de usuário são aceitas neste campo.
Protocolo de privacidade	Select privacy protocol - DES or AES.
Frase privada	Entre om a frase privada. Macros de usuário são aceitas neste campo.

Note:

Desde o Zabbix 2.2, os protocolos SHA e AES são suportados para a autenticação e privacidade SNMPv3, em adição ao MD5 e ao DES que já eram suportados.

Exemplo 1

Exemplo geral:

Parâmetro	Descrição
Comunidade	public
SNMP OID	1.2.3.45.6.7.8.0 (ou .1.2.3.45.6.7.8.0)
Chave	<Texto único que será utilizado como referência nas triggers> Por exemplo: "my_param".

Observe que o **SNMP OID** poderá ser o número ou seu nome. Em alguns casos pode ser que o Zabbix não consiga resolver o número a partir do nome, para estas situações recomenda-se utilizar o 'snmpget' para fazer esta resolução antes de configurar o item:

```
shell> snmpget -On localhost public enterprises.ucdavis.memory.memTotalSwap.0
```

Monitoração através de SNMP só é possível se, durante a configuração do binário do Zabbix Server ou Zabbix Proxy, tiver sido utilizado o parâmetro --with-net-snmp.

Exemplo 2

Monitorando o Uptime (Disponibilidade):

Parâmetro	Descrição
OID	MIB::sysUpTime.0
Key	router.uptime
Value type	Float
Units	uptime
Multiplier	0.01

Funcionamento interno do processamento em lote

A partir do Zabbix 2.2.3 os processos do Zabbix Server e Zabbix Proxy tem a capacidade de solicitar vários dados através de SNMP em uma única requisição. Esta característica afeta diversos tipos de itens SNMP:

- itens SNMP normais;
- Itens SNMP com índice dinâmico;
- Regras de autobusca SNMP.

Todos os itens SNMP, com igual intervalo de coletas, são agendados para serem coletados juntos. Para os primeiros dois tipos de itens a fila agrupa até 128 itens ao mesmo tempo, enquanto as regras de autobusca são processadas de forma individual.

No nível mais baixo, dois tipos de operações são feitas para obter os valores: obter múltiplos objetos e varrer árvore OID.

Para a ação de "obtenção", uma requisição 'GetRequest-PDU' é utilizada com até 128 solicitações. Para a "varredura", uma requisição "GetNextRequest-PDU" será utilizada com o SNMPv1 e uma 'GetBulkRequest' com "max-repetitions" definido como 128 será utilizada com SNMPv2 e SNMPv3.

Os benefícios de se utilizar o processamento em lote de grandes volumes de itens SNMP são listados a seguir:

- itens SNMP normais se beneficiam de uma "obtenção" otimizada;
- itens SNMP com índices dinâmicos se beneficiam tanto da "obtenção" quanto da "varredura" otimizada: "obtenção" é utilizada para a verificação de índice e a "varredura" para construir o cache;
- regras de autobusca SNMP se beneficiam da melhoria na "varredura".

Entretanto, existem limitações técnicas que fazem com que nem todos os dispositivos sejam capazes de retornar 128 valores por requisição. Algumas vezes o dispositivo pode retornar com a mensagem "(1) tooBig" ou não responder a todos uma vez que a quantidade é superior ao limite do equipamento.

Para definir o máximo de objetos para se solicitar a um dispositivo o Zabbix In order to find an optimal number of objects to query for a given device, Zabbix uses the following strategy: * A primeira requisição solicita 1 valor apenas; * Se obteve sucesso, solicita 2 valores; * Se obteve sucesso, solicita 3 valores; * Se obteve sucesso, solicita 4 valores; Obtendo sucesso nas tentativas as próximas consultas requisitarão, respectivamente: 6, 9, 13, 19, 28, 42, 63, 94, 128 (multiplicação do último valor bem sucedido por 1,5).

Uma vez que o dispositivo não consiga dar uma resposta adequada a uma quantidade de itens simultâneos (por exemplo, 42 valores), o Zabbix faz duas coisas.

Primeiro, ele reduz o tamanho do lote que falhou pela metade (no caso 21). Se o dispositivo responder de forma adequada, será feito o ajuste para 28 itens (fator anterior). Entretanto, se a falha persistir, o Zabbix voltará a solicitar os itens 1 por 1. Se o dispositivo voltar a falhar, o problema não será de quantidade de valores simultâneos.

A segunda coisa que ele fará para os lotes de itens subsequentes é começar com o último número de valores obtidos com sucesso (28 em nosso exemplo) e continuará solicitando valores somando 1 ao limite de requisições simultâneas até encontrar o limite. Quando descobrir o limite exato o Zabbix não mais fará estes testes e passará a trabalhar somente com o limite descoberto neste dispositivo.

Se grandes consultas falharem com esta quantidade de valores solicitados, pode significar uma de duas coisas. Os critérios exatos que o dispositivo utiliza para limitar a quantidade de valores não pode ser descoberta, mas tentamos aproximar o valor. Assim a primeira possibilidade é que este é o número de valores simultâneos que representa algo próximo do limite real pois, normalmente, a resposta é menor do que o limite. A segunda possibilidade é que um pacote UDP, em qualquer direção, simplesmente se perdeu. Por estas razões o Zabbix considera que a consulta falhou e reduz o limite de valores simultâneos para tentar trabalhar numa faixa mais confortável para o dispositivo mas, (a partir do 2.2.8) somente até duas tentativas.

No exemplo acima, se uma consulta com 32 requisições simultâneas falhar, o Zabbix irá reduzir o contador para 31. Se falhar novamente, reduz para 30. Entretanto, o Zabbix não irá reduzir de 30 pois entenderá que a falha se deve a perda de pacotes UDP e não por limite do dispositivo.

Se um dispositivo não conseguir trabalhar com requisições em lote por outras razões, e a eurística acima não funcionar, desde o Zabbix 2.4 existe uma opção chamada "Usar requisições em lote" em cada interface que poderá ser desativada.

1 Índices Dinâmicos

Visão Geral

Embora possa ter encontrado o índice do item que deseja (por exemplo, uma interface de rede) entre os diversos OIDs SNMP você deve certificar que trata-se de um índice fixo, não assumindo com apenas uma pesquisa, que o número permanecerá sempre o mesmo.

Os índices podem ser dinâmicos, fazendo com que mudem após algum tempo e o monitoramento do item pare de funcionar.

Para evitar este cenário, é possível definir um OID que faz referência a um índice com possibilidade de sofrer alteração.

Por exemplo, se você precisa identificar o valor do índice que corresponda a interface **GigabitEthernet0/1** em um equipamento Cisco para adicionar em **ifInOctets**, use o OID a seguir:

```
ifInOctets["index","ifDescr","GigabitEthernet0/1"]
```

A sintaxe

Uma sintaxe especial é utilizada para estas consultas:

<OID of data>["índice","<OID base>","<texto a procurar>"]

Parameter	Description
OID base Índice	OID principal para pesquisar o dado que se deseja o índice. Método de processamento. Atualmente é suportado apenas um método: índice - procura pelo índice para adicionar ao OID Base
OID base do índice	Este OID será utilizado junto com a informação de índice para recuperar o valor solicitado.
Texto a procurar	Texto utilizado para localizar o valor que se deseja. Sensitivo ao caso.

Exemplo

Recuperando uso de memória do processo do apache.

Se utilizar esta sintaxe de OID:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem["index", "HOST-RESOURCES-MIB::hrSWRunPath", "/usr/sbin/apache2"]
```

O número do índice será localizado aqui:

```
...
HOST-RESOURCES-MIB::hrSWRunPath.5376 = STRING: "/sbin/getty"
HOST-RESOURCES-MIB::hrSWRunPath.5377 = STRING: "/sbin/getty"
HOST-RESOURCES-MIB::hrSWRunPath.5388 = STRING: "/usr/sbin/apache2"
HOST-RESOURCES-MIB::hrSWRunPath.5389 = STRING: "/sbin/sshd"
...
```

Neste caso nós encontramos o índice **5388**. Este número será adicionado ao **OID base** para recuperar o valor que estamos interessados:

```
HOST-RESOURCES-MIB::hrSWRunPerfMem.5388 = INTEGER: 31468 KBytes
```

Cache de índices

Quando um índice dinâmico é solicitado, o Zabbix recupera e faz o cache da tabela SNMP inteira pesquisada. Isso é feito de forma a acelerar caso outro índice necessite de dados da mesma tabela. Observe que cada 'poller' do Zabbix mantém um cache em separado.

Para todos os valores subsequentes, apenas o índice recuperado será utilizado. Se ele não tiver sido alterado, solicita-se o valor desejado. Se tiver sido alterado, a tabela SNMP será solicitada novamente.

2 Special OIDs

Alguns dos OIDs SNMP mais usados são traduzidos automaticamente para uma representação numérica pelo Zabbix. Por exemplo, **ifIndex** é traduzida para **1.3.6.1.2.1.2.2.1.1**, **ifIndex.0** é traduzida para **1.3.6.1.2.1.2.2.1.1.0**.

A tabela abaixo contém a lista dos OIDs especiais.

Special OID	Identifier	Description
ifIndex	1.3.6.1.2.1.2.2.1.1	Valor único para cada interface.
ifDescr	1.3.6.1.2.1.2.2.1.2	Um texto contendo informações sobre a interface. Deve incluir o nome do fabricante, nome do produto e versão da interface.
ifType	1.3.6.1.2.1.2.2.1.3	O tipo de interface, distinguido de acordo com o(s) protocolo(s) físico/link imediatamente 'abaixo' da camada de rede na pilha de protocolos.
ifMtu	1.3.6.1.2.1.2.2.1.4	O tamanho máximo do datagrama que pode ser enviado/recebido na interface, especificado em octetos.
ifSpeed	1.3.6.1.2.1.2.2.1.5	Uma estimativa da largura de banda atual da interface em bits por segundo.

Special OID	Identifier	Description
ifPhysAddress	1.3.6.1.2.1.2.2.1.6	O endereço da interface na camada de protocolo imediatamente 'abaixo' da camada de rede na pilha de protocolos.
ifAdminStatus	1.3.6.1.2.1.2.2.1.7	O estado administrativo atual da interface.
ifOperStatus	1.3.6.1.2.1.2.2.1.8	O estado operacional atual da interface.
ifInOctets	1.3.6.1.2.1.2.2.1.10	O número total de octetos recebidos na interface, incluindo caracteres de enquadramento.
ifInUcastPkts	1.3.6.1.2.1.2.2.1.11	O número de pacotes unicast de sub-redes entregues a um protocolo de camada superior.
ifInNUcastPkts	1.3.6.1.2.1.2.2.1.12	O número de pacotes não unicast (ou seja, sub-rede multicast ou sub-rede broadcast) entregues a um protocolo de camada superior.
ifInDiscards	1.3.6.1.2.1.2.2.1.13	O número de pacotes de entrada que foram escolhidos para serem descartados mesmo que nenhum erro tenha sido detectado para evitar que sejam entregues a um protocolo de camada superior. Um possível motivo para descartar um pacote desse tipo pode ser liberar espaço no buffer.
ifInErrors	1.3.6.1.2.1.2.2.1.14	O número de pacotes de entrada que continham erros impedindo que fossem entregues a um protocolo de camada superior.
ifInUnknownProtos	1.3.6.1.2.1.2.2.1.15	O número de pacotes recebidos pela interface e que foram descartados devido a um protocolo desconhecido ou não suportado.
ifOutOctets	1.3.6.1.2.1.2.2.1.16	O número total de octetos transmitidos para fora da interface, incluindo caracteres de enquadramento.
ifOutUcastPkts	1.3.6.1.2.1.2.2.1.17	O número total de pacotes que os protocolos de nível superior solicitaram que fossem transmitidos e que não foram endereçados a um endereço multicast ou broadcast nesta subcamada, incluindo aqueles que foram descartados ou não enviados.
ifOutNUcastPkts	1.3.6.1.2.1.2.2.1.18	O número total de pacotes que os protocolos de nível superior solicitaram que fossem transmitidos e que foram endereçados a um endereço multicast ou broadcast nesta subcamada, incluindo aqueles que foram descartados ou não enviados.
ifOutDiscards	1.3.6.1.2.1.2.2.1.19	O número de pacotes de saída que foram escolhidos para serem descartados mesmo que nenhum erro tenha sido detectado para impedir sua transmissão. Um possível motivo para descartar um pacote desse tipo pode ser liberar espaço no buffer.
ifOutErrors	1.3.6.1.2.1.2.2.1.20	O número de pacotes de saída que não puderam ser transmitidos devido a erros.
ifOutQLen	1.3.6.1.2.1.2.2.1.21	O comprimento da fila de pacotes de saída (em pacotes).

3 Arquivos MIB

Intrudução

MIB é um acrônimo de Management Information Base. Os arquivos MIB permitem que você use a representação textual do OID (Object Identifier).

Por exemplo,

`ifHCOutOctets`

a representação textual do OID é

`1.3.6.1.2.1.31.1.1.1.10`

Você pode usar ambos ao monitorar dispositivos SNMP com o Zabbix, mas se sentir mais confortável ao usar a representação textual será necessário instalar os arquivos MIB.

Instalando arquivos MIB

Sistemas baseados em Debian:

```
# apt install snmp-mibs-downloader
# download-mibs
```

Sistemas baseados em RedHat:

```
# yum install net-snmp-libs
```

Habilitando arquivo MIB

Em sistemas baseados em RedHat os arquivos MIB devem ser habilitados por padrão. Em sistemas baseados em Debian você tem que editar o arquivo `/etc/snmp/snmp.conf` e comente a linha que possui `mibs` :

```
# As the snmp packages come without MIB files due to license reasons, loading
# of MIBs is disabled by default. If you added the MIBs you can re-enable
# loading them by commenting out the following line.
#mibs :
```

Testando os arquivos MIB

O teste de MIBs SNMP pode ser feito usando o utilitário `snmpwalk`. Para instalar use as seguintes instruções.

Sistemas baseados em Debian:

```
# apt install snmp
```

Sistemas baseados em RedHat:

```
# yum install net-snmp-utils
```

Depois disso, o comando a seguir não deve apresentar erro ao consultar um dispositivo de rede:

```
$ snmpwalk -v 2c -c public <NETWORK DEVICE IP> ifInOctets
IF-MIB::ifInOctets.1 = Counter32: 176137634
IF-MIB::ifInOctets.2 = Counter32: 0
IF-MIB::ifInOctets.3 = Counter32: 240375057
IF-MIB::ifInOctets.4 = Counter32: 220893420
[...]
```

Usando MIBs no Zabbix

O mais importante a ter em mente é que os processos do Zabbix não são informados sobre as alterações feitas nos arquivos MIB. Então, depois de cada mudança você deve reiniciar o Zabbix server ou Proxy:

```
# service zabbix-server restart
```

Depois disso, as alterações feitas nos arquivos MIB entraram em vigor.

Usando arquivos MIB personalizados

Existem arquivos MIB padrão que vem com todas as distribuições GNU/Linux. Mas alguns fornecedores de dispositivos fornecem seus próprios arquivos.

Digamos que você gostaria de usar o arquivo MIB **CISCO-SMI**. As instruções a seguir irão baixar e instalar o arquivo MIB:

```
# wget ftp://ftp.cisco.com/pub/mibs/v2/CISCO-SMI.my -P /tmp
# mkdir -p /usr/local/share/snmp/mibs
# grep -q '^mibdirs +/usr/local/share/snmp/mibs' /etc/snmp/snmp.conf 2>/dev/null || echo "mibdirs +/usr/local/share/snmp/mibs"
# cp /tmp/CISCO-SMI.my /usr/local/share/snmp/mibs
```

Agora você deve ser capaz de usá-lo. Tente traduzir o nome do objeto ciscoProducts do arquivo MIB para OID:

```
# snmptranslate -IR -On CISCO-SMI::ciscoProducts
.1.3.6.1.4.1.9.1
```

Se você receber erros em vez do OID, certifique-se de que todos os comandos foram executados sem erros.

Se tradução do nome do objeto funcionou, você está pronto para usar o arquivo MIB personalizado. Observe o prefixo do nome MIB (CISCO-SMI::) usado na consulta. Você precisará disso ao usar uma ferramenta de linha de comando, bem como o Zabbix.

Não se esqueça de reiniciar o Zabbix server ou Proxy antes de usar o arquivo MIB no Zabbix.

::: Nota importante Tenha em mente que os arquivos MIB podem ter dependências. Ou seja, um MIB pode exigir outro. Para satisfazer essas dependências você tem que instalar todos os arquivos MIBs. :::

3 Traps SNMP

Visão geral

Receber traps SNMP é o oposto de consultar dispositivos habilitados para SNMP.

Nesse caso, as informações são enviadas de um dispositivo habilitado para SNMP e são coletados ou "trapped" pelo Zabbix

Normalmente, os traps são enviados mediante mudança de condição e o agente conecta-se ao servidor na porta 162 (em oposição à porta 161 no lado do agente que é usado para consultas). O uso de traps pode detectar alguns pequenos problemas que ocorrem durante o intervalo de consulta e podem ser perdidos pelo dado consultado.

O recebimento de traps SNMP no Zabbix foi projetado para funcionar com **snmptrapd** e um dos mecanismos embutidos para passar as traps para o Zabbix - um script Perl ou SNMPPTT.

O fluxo de trabalho de recebimento de uma trap:

1. **snmptrapd** recebe uma trap
2. snmptrapd passa a trap para SNMPPTT ou chama o receptor de Perl trap
3. O receptor SNMPPTT or Perl trap analisa, formata e escreve a trap em um arquivo
4. O Zabbix SNMP trapper lê e analisa o arquivo
5. Para cada trap o Zabbix encontra todos os itens "SNMP trapper" com a interface do host que correspondem ao endereço da trap recebida. Observe que apenas o "IP" ou "DNS" na interface do host é usado durante a análise.
6. Para cada item encontrado, a trap é comparada em "snmptrap[regexp]". A trap é definida como o valor **all** nos itens correspondentes. Se nenhum item correspondente for encontrado e houver um item "snmptrap.fallback", o trap é definido com esse valor.
7. Se o trap não foi definido como valor de nenhum item, o Zabbix por padrão registra a trap sem correspondência. (Isso é configurado como "Registrar traps SNMP não correspondentes" em Administração → Geral → Outros.)

1 Configurando traps SNMP

A configuração dos seguintes campos no frontend é específica para Tipos de itens:

- Seu host deve ter uma interface SNMP

Em Configuração→ Hosts, no campo **Interface do host** defina uma interface SNMP com o endereço IP ou DNS correto. o endereço de cada interceptação recebida é comparada com os endereços IP ou DNS de todos os endereços das interfaces SNMP para encontrar os hosts correspondentes.

- Configurar o item

No campo **Chave**, use uma das chaves trap SNMP:

Chave		
Descrição	Valor de retorno	Comentários
snmptrap[regexp]		

Chave		
Captura todas as traps SNMP que correspondem à expressão regular especificada em regexp . Se regexp não for especificado, é capturado qualquer trap.	Trap SNMP	Este item pode ser definido apenas para interfaces SNMP Este item é suportado desde o Zabbix 2.0.0 . Nota: Iniciando com o Zabbix 2.0.5, macros de usuário e expressões regulares globais são suportadas no parâmetro desta chave de item.
snmptrap.fallback Captura todas as traps SNMP que não foram capturadas por nenhum dos itens snmptrap[] dessa interface.	SNMP trap	Este item é suportado desde o Zabbix 2.0.0 .

Note:

Expressões regulares de várias linhas não são suportadas neste momento

Defina o **Tipo de informação** como 'Log' para que os registros de hora sejam processados. Observe que qualquer outro formato, tal qual o numérico, também são aceitáveis mas requerem um gerenciado de trap personalizado.

Note:

Para a monitoração de trap SNMP, você primeiro deve configurá-lo corretamente.

2 Configurando a monitoração via trap SNMP

Configurando o Zabbix Server/Proxy

Para receber as traps, o Zabbix Server/Proxy deverá estar configurado para iniciar o processo de SNMP trapper e apontado para o arquivo de traps que estará sendo alimentado pelo SNMPTT ou pelo 'perl trap receiver'. Para fazer isso, edite o arquivo de configuração (**zabbix_server.conf** ou **zabbix_proxy.conf**):

1. StartSNMPTrapper=1
2. SNMPTTrapperFile=[TRAP FILE]

Warning:

Se o parâmetro do systemd **PrivateTmp** tiver sido definido é improvável que funcione no /tmp.

Configurando o SNMPTT

Primeiramente o 'snmptrapd' precisa estar configurado para utilizar o SNMPTT.

Note:

Para uma melhor performance, o SNMPTT precisa estar configurado como um daemon utilizando o **snmptthandler-embedded** para enviar as traps. Veja mais instruções sobre configuração do SNMPTT neste endereço:
<http://snmptt.sourceforge.net/docs/snmptt.shtml>

Quando o SNMPTT estiver apto a receber as 'traps', ajuste-o para registrar as traps:

1. registre as traps no arquivo de traps que será lido pelo Zabbix:
log_enable = 1
log_file = [TRAP FILE]
2. defina o formato de data e hora:
date_time_format = %H:%M:%S %Y/%m/%d = [DATE TIME FORMAT]

Agora formate as traps de forma a possibilitar que o Zabbix as reconheça (edite o arquivo snmptt.conf):

1. Cada instrução de formato deverá começar com "ZBXTRAP [address]", onde [address] será o valor usado na comparação com as interfaces (pelo IP ou DNS). Exemplo.:
EVENT coldStart .1.3.6.1.6.3.1.1.5.1 "Status Events" Normal
FORMAT ZBXTRAP \$aA Device reinitialized (coldStart)

2. Veja mais sobre formato de SNMP trap a seguir

Attention:

Não utilize traps 'não esperadas' - o Zabbix não estará apto a reconhecê-las. Traps não esperadas podem ser tratadas ao definir um evento geral no `snmptt.conf`:
EVENT general .* "General event" Normal

Configurando o Perl trap receiver

Requerimentos: Perl, pacote Net-SNMP compilado com `--enable-embedded-perl` (já é padrão no Net-SNMP 5.4)

'Perl trap receiver' (olhe em `misc/snmptrap/zabbix_trap_receiver.pl`) poderá ser utilizado para encaminhar as traps para o Zabbix Server/Proxy diretamente a partir do `snmptrapd`. Para configurar:

- adicione o script perl no arquivo de configuração do 'snmptrapd' (`snmptrapd.conf`), Exemplo:
perl do "[FULL PATH TO PERL RECEIVER SCRIPT]";
- configure o receptor, exemplo:
`$SNMPTrapperFile = '[TRAP FILE]';`
`$DateTimeFormat = '[DATE TIME FORMAT]';`

Note:

Se o nome do script não estiver entre aspas, o `snmptrapd` irá se recusar a iniciar com mensagens similares a estas:
Regex modifiers `"/l"` and `"/a"` are mutually exclusive at (eval 2) line 1, at end of line
Regex modifier `"/l"` may not appear twice at (eval 2) line 1, at end of line

Formato do SNMP trap

Tanto os 'perl trap receivers' quanto o 'SNMPTT trap configuration' precisam formatar a trap conforme o padrão a seguir: **[timestamp] [the trap, part 1] ZBXTRAP [address] [the trap, part 2]**, onde

- [timestamp] - momento de ocorrência do evento
- ZBXTRAP - cabeçalho que indica o início de uma nova linha de trap
- [address] - endereço IP para localizar o host a receber a trap

Observe que o "ZBXTRAP" e o "[address]" serão removidos da mensagem durante o processamento. Se a trap utilizar outro formato, o Zabbix pode analisá-las de forma inesperada.

Exemplo de trap:

11:30:15 2011/07/27 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events" localhost - ZBXTRAP 192.168.1.1 Link down on interface 2.
Admin state: 1. Operational state: 2

Esta trap será enviada para um host com a interface SNMP com IP=192.168.1.1:

11:30:15 2011/07/27 .1.3.6.1.6.3.1.1.5.3 Normal "Status Events" localhost - Link down on interface 2. Admin state: 1.

3 Requerimentos de sistema

Rotação de logs

O Zabbix não provê rotação de logs que possa ser gerida pelo usuário. A rotação de logs primeiro irá renomear o arquivo antigo e após isso apaga-lo para garantir que nenhuma trap será perdida, conforme processo a seguir:

1. O Zabbix abre o arquivo de traps na última posição conhecida e vai para o passo 3
2. O Zabbix verifica se o arquivo aberto já foi rotacionado ao comparar o número do inode definido para o mesmo. Se não existe arquivo aberto, o Zabbix reseta a informação de última posição conhecida e retorna ao passo 1.
3. O Zabbix lê o dado do arquivo aberto e define uma nova localização.
4. Os novos dados são analisados. Se era o arquivo rotacionado, o arquivo é fechado e retorna ao passo 2.
5. Se não existe novo dado, o Zabbix aguarda 1 segundo e retorna ao passo 2.

Attention:

O tamanho máximo de um arquivo de log para o Zabbix é de 2GB. O arquivo de log precisa ser rotacionado antes deste limite.

Sistema de arquivos

Devido à forma de implementação, o Zabbix precisa que o sistema de arquivos suporte inodes para diferenciar os arquivos (a informação é obtida através da função `stat()`).

4 Exemplo de configuração

Este exemplo utiliza o `snmptrapd` + `SNMPTT` para enviar traps ao Zabbix Server:

1. **zabbix_server.conf** - configure o Zabbix para iniciar o SNMP trapper e defina a localização do arquivo file:
StartSNMPTrapper=1
SNMPTrapperFile=/tmp/my_zabbix_traps.tmp
2. **snmptrapd.conf** - adicione o `SNMPTT` como o gerenciador de traps:
traphandle default snmptt
3. **snmptt.ini** - configure o arquivo de saída e o formato de hora:
log_file = /tmp/my_zabbix_traps.tmp
date_time_format = %H:%M:%S %Y/%m/%d
4. **snmptt.conf** - defina o formato padrão de trap:
EVENT general .* "General event" Normal
FORMAT ZBXTRAP \$aA \$ar
5. Crie um item SNMP TEST:
IP da Interface SNMP do Host IP: 127.0.0.1
Chave: snmptrap["General"]
Formato de hora do log: hh:mm:ss yyyy/MM/dd

Teste o funcionamento:

1. Comando utilizado para enviar uma trap:
snmptrap -v 1 -c public 127.0.0.1 '.1.3.6.1.6.3.1.1.5.3' '0.0.0.0' 6 33 '55' .1.3.6.1.6.3.1.1.5.3 s "teststring000"
2. Que será recebida:
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - ZBXTRAP 127.0.0.1 127.0.0.1
3. Valor par ao teste do item:
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - 127.0.0.1

Note:

Este exemplo simples utiliza o `SNMPTT` como **traphandle**. Para melhor performance em ambientes de produção, utilize um script Perl para encaminhar as traps do `snmptrapd` ao `SNMPTT` ou diretamente ao Zabbix.

5 Setup example

This example uses `snmptrapd` + `SNMPTT` to pass traps to Zabbix server. Setup:

1. **zabbix_server.conf** - configure Zabbix to start SNMP trapper and set the trap file:
StartSNMPTrapper=1
SNMPTrapperFile=/tmp/my_zabbix_traps.tmp
2. **snmptrapd.conf** - add `SNMPTT` as the trap handler:
traphandle default snmptt
3. **snmptt.ini** -
enable the use of the Perl module from the NET-SNMP package:
net_snmp_perl_enable = 1
configure output file and time format:
log_file = /tmp/my_zabbix_traps.tmp
date_time_format = %H:%M:%S %Y/%m/%d
4. **snmptt.conf** - define a default trap format:
EVENT general .* "General event" Normal
FORMAT ZBXTRAP \$aA \$ar
5. Create an SNMP item TEST:
Host's SNMP interface IP: 127.0.0.1
Key: snmptrap["General"]
Log time format: hh:mm:ss yyyy/MM/dd

This results in:

1. Command used to send a trap:
snmptrap -v 1 -c public 127.0.0.1 '.1.3.6.1.6.3.1.1.5.3' '0.0.0.0' 6 33 '55' .1.3.6.1.6.3.1.1.5.3 s "teststring000"
2. The received trap:
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - ZBXTRAP 127.0.0.1 127.0.0.1
3. Value for item TEST:
15:48:18 2011/07/26 .1.3.6.1.6.3.1.1.5.3.0.33 Normal "General event" localhost - 127.0.0.1

Note:

This simple example uses `SNMPTT` as **traphandle**. For better performance on production systems, use embedded Perl to pass traps from `snmptrapd` to `SNMPTT` or directly to Zabbix.

6 See also

- [Zabbix blog article on SNMP traps](#)
- [CentOS based SNMP trap tutorial on zabbix.org](#)

4 Verificações IPMI

Visão geral

Você pode monitorar a saúde e disponibilidade de dispositivos IPMI (Interface Inteligente de Gerenciamento de Plataforma) no Zabbix. Para executar verificações IPMI o Zabbix Server deve ser inicialmente **configurado** com suporte a IPMI.

IPMI é uma interface padronizada para gerenciamento remoto "fora de banda" (OOB) e "luzes apagadas" (LOM) de sistemas de computador. Ela permite monitorar o estado do hardware diretamente das chamadas placas de gerenciamento fora de banda, independentemente do sistema operacional ou mesmo se a máquina está ou não ligada.

O monitoramento IPMI do Zabbix funciona apenas com dispositivos que possuem suporte a IPMI (HP iLO, DELL DRAC, IBM RSA, Sun SSP, etc.).

Desde o Zabbix 3.4, um novo processo gerenciador de IPMI foi adicionado às verificações IPMI agendadas por sondagem IPMI. Agora um host é sempre sondado por apenas um sondador IPMI por vez, reduzindo o número de conexões abertas para controladores BMC. Com essas mudanças é seguro aumentar o número de sondadores IPMI sem se preocupar com sobrecarga de controlador BMC. O processo gerenciador de IPMI é automaticamente iniciado quando ao menos um sondador IPMI é iniciado.

Veja também **problemas conhecidos** para verificações IPMI.

Configuração

Configuração de Host

Um host deve estar configurado para processar verificações IPMI. Uma interface IPMI deve ser adicionada, com os respectivos números de IP e porta, e os parâmetros de autenticação IPMI devem estar definidos.

Consulte a **configuração de hosts** para mais detalhes.

Configuração do Server

Por padrão, o Zabbix Server não é configurado para iniciar qualquer sondador IPMI, portanto quaisquer itens IPMI adicionados não funcionarão. Para mudar isto, abra o arquivo de configuração do Zabbix Server (**zabbix_server.conf**) como root e procure pela seguinte linha:

```
# StartIPMIPollers=0
```

Descomente esta linha e configure a quantidade de sondadores para, digamos, 3, ficando então desta forma:

```
StartIPMIPollers=3
```

Salve o arquivo e reinicie o zabbix_server em seguida.

Configuração de item

Quando **configurando um item** no nível de host:

- Selecione 'Agente IPMI' como o Tipo
- Informe uma **chave** de item única dentro do host (digamos, ipmi.fan.rpm)
- Para Interface de host selecione a interface IPMI relevante (IP e porta). Note que uma interface IPMI deve existir no host.
- Defina o Sensor IPMI (por exemplo 'FAN MOD 1A RPM' no Dell Poweredge) do qual obter as métricas. Por padrão, o ID do sensor deve ser especificado. É também possível usar prefixos antes do valor:
 - **id:** - para especificar o ID do sensor;
 - **name:** - para especificar o nome completo do sensor. Isto pode ser útil em situação em que os sensores só podem ser distinguidos especificando o nome completo.
- Selecione o tipo de informação respectivo ('Numérico (número flutuante)' neste caso; para sensores discretos - 'Numérico (unsigned)'), units (unidades - provavelmente 'rpm') e quaisquer outros atributos de item necessários

Verificações suportadas

A tabela abaixo descreve itens nativos que são suportados nas verificações IPMI do agente.

Chave de item

▲	Descrição	Valore de retorno	Comentários
ipmi.get			

Informação rela- cionada ao sensor IPMI.	Objeto JSON	Este item pode ser usado para de- scoberta de sensores IPMI. Suportado desde o Zabbix 5.0.0.
--	----------------	---

Tempo limite (timeout) e término de sessão

O tempo limite das mensagens IPMI e contagem de novas tentativas são definidos na biblioteca OpenIPMI. Devido ao desenho atual da OpenIPMI, não é possível tornar estes valores configuráveis no Zabbix, nem na interface e nem em nível de item.

A expiração de sessão IPMI por inatividade é de 60 +/-3 segundos para LAN. Atualmente não é possível implementar envio periódico de comando Activate Session com OpenIPMI. Se não houver verificações de item IPMI do Zabbix para um BMC particular por mais do que o tempo limite de sessão configurado no BMC, então a próxima verificação IPMI depois da expiração de tempo irá expirar devido ao tempo limite individual das mensagens, novas tentativas ou recebimento de erro. Após isso, uma nova sessão é aberta e é iniciada uma releitura completa do BMC. Se você quiser evitar releituras desnecessárias do BMC é aconselhável configurar o intervalo de sondagem do item para um valor abaixo ao tempo limite de sessão do IPMI por inatividade configurado no BMC.

Notas sobre sensores discretos do IPMI

Para encontrar sensores em um host inicie o Zabbix Server com **DebugLevel=4** habilitado. Aguarde alguns minutos e encontre os registros de descoberta de sensores no arquivo de log do Zabbix Server:

```
$ grep 'Added sensor' zabbix_server.log
8358:20130318:111122.170 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:7 id:'CATERR' reading_type:
8358:20130318:111122.170 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15 id:'CPU Therm Trip' read
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'System Event Log' re
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'PhysicalSecurity' re
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'IPMI Watchdog' readi
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'Power Unit Stat' rea
8358:20130318:111122.171 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'P1 Therm Ctrl %' rea
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:16 id:'P1 Therm Margin' rea
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13 id:'System Fan 2' readin
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:13 id:'System Fan 3' readin
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'P1 Mem Margin' readi
8358:20130318:111122.172 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'Front Panel Temp' re
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:15 id:'Baseboard Temp' read
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB +5.0V' reading_typ
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'BB +3.3V STBY' readi
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:9 id:'BB +3.3V' reading_typ
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'BB +1.5V P1 DDR3' re
8358:20130318:111122.173 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:17 id:'BB +1.1V P1 Vccp' re
8358:20130318:111122.174 Added sensor: host:'192.168.1.12:623' id_type:0 id_sz:14 id:'BB +1.05V PCH' readi
```

Para compreender os tipos e estado dos sensores, acesse uma cópia das especificações IPMI 2.0 em <http://www.intel.com/content/www/us/en/servers/ipmi/ipmi-specifications.html> (No momento da documentação o material mais recente era <http://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/second-gen-interface-spec-v2.pdf>, sendo que não foi encontrada versão em Português Brasileiro)

O primeiro parâmetro com o qual começar é o "reading_type". Use a tabela "Table 42-1, Event/Reading Type Code Ranges" das especificações para interpretar o código "reading_type" (tipo de leitura). A maioria dos sensores no nosso exemplo possui "reading_type:0x1" o que significa sensor de limite, a tabela "Table 42-3, Sensor Type Codes" mostra que "type:0x1" significa sensor de temperatura, "type:0x2" - sensor de tensão, "type:0x4" - Cooler, etc. Sensores de limite algumas vezes são chamados de "analógicos" por medirem parâmetros contínuos como temperatura, tensão, revoluções por minuto.

Outro exemplo - um sensor com "reading_type:0x3". A tabela "Table 42-1, Event/Reading Type Code Ranges" diz que códigos de leitura 02h-0Ch significam sensor "Discreto Genérico". Sensores discretos possuem até 15 estados possíveis (em outras palavras - até 15 bits significativos). Por exemplo, para sensor 'CATERR' com "type:0x7" a "Table 42-3, Sensor Type Codes" mostra que este

tipo significa "Processor" (Processador) e o significado dos bits individuais é: 00h (o último bit significante) - IERR, 01h - Thermal Trip, etc.

Há uns poucos sensores com "reading_type:0x6f" em nosso exemplo. Para estes sensores a tabela "Table 42-1, Event/Reading Type Code Ranges" aconselha usar a "Table 42-3, Sensor Type Codes" para interpretar os bits significativos. Por exemplo, o sensor 'Power Unit Stat' tem o tipo "type:0x9" que significa "Power Unit" (Unidade de Potência). O deslocamento (offset) 00h significa "PowerOff/Power Down" (Desligar). Em outras palavras se o último bit significativo é 1, então o servidor está desligado. Para testar este bit, pode ser usada a **função bitand** com máscara '1'. A expressão de gatilho poderia ser algo como:

```
bitand(last(/www.example.com/Power Unit Stat,#1),1)=1
```

para alertar sobre o desligamento de um servidor.

Notas sobre nomes de sensores discretos em OpenIPMI-2.0.16, 2.0.17, 2.0.18 e 2.0.19

Nomes de sensores discretos em OpenIPMI-2.0.16, 2.0.17 e 2.0.18 frequentemente possuem um "0" adicional (ou algum outro dígito ou letra) na parte final. Por exemplo, enquanto ipmitool e OpenIPMI-2.0.19 mostram o nome de sensor como "PhysicalSecurity" ou "CATERR", em OpenIPMI-2.0.16, 2.0.17 e 2.0.18 os nomes seriam "PhysicalSecurity0" ou "CATERR0", respectivamente.

Quando configurando um item IPMI com Zabbix Server usando OpenIPMI-2.0.16, 2.0.17 e 2.0.18, use terminações de nome com "0" no campo sensor IPMI dos itens de agente IPMI. Quando o seu Zabbix Server for atualizado para uma nova distribuição Linux, que usa OpenIPMI-2.0.19 (ou mais recente), os itens com estes sensores IPMI discretos se tornarão "NÃO SUPORTADOS". Você terá que alterar seus nomes de sensor IPMI (remova o '0' do final) e aguardar um momento até que eles se tornem "Habilitados" novamente.

Notas sobre disponibilidade simultânea de sensor de limite (threshold) e sensor discreto

Alguns agentes IPMI oferecem suporte a um sensor de limite e um sensor discreto sob o mesmo nome. Nas versões de Zabbix anteriores a 2.2.8 e 2.4.3, o primeiro sensor fornecido era escolhido. Desde as versões 2.2.8 e 2.4.3, a preferência é sempre dada ao sensor de limite.

Notas sobre encerramento de conexão

Se as verificações IPMI não forem executadas (por qualquer razão: todos os itens IPMI desabilitados/não suportados, host desabilitado/deletado, host em manutenção, etc.) a conexão IPMI será encerrada no Zabbix Server ou Proxy dentro de 3 a 4 horas, dependendo do horário em que o Zabbix Server/Proxy foi iniciado.

5 Verificações simples

Visão geral

Verificações simples são normalmente usadas para verificações remotas de serviço, sem agente.

Note que o Zabbix Agent não é necessário para verificações simples. O Zabbix Server/Proxy é responsável pelo processamento das verificações simples (fazendo conexões externas, etc.).

Exemplos de uso de verificações simples:

```
net.tcp.service[ftp,,155]
net.tcp.service[http]
net.tcp.service.perf[http,,8080]
net.udp.service.perf[ntp]
```

Note:

Os campos Nome de usuário e Senha na configuração de um item de verificação simples são usados para monitoramento de itens VMware; são ignorados de outra forma.

Verificações simples suportadas

Lista de verificações simples suportadas:

Veja também:

- [Chaves de item de monitoramento VMware](#)

Chave

Descrição	Valore de retorno	Parâmetros	Comentários
icmping[<target>,<packets>,<interval>,<size>,<timeout>]			

	Acessibilidade de host por ping ICMP.	0 - falha de ping ICMP 1 - ping ICMP bem-sucedido	target - IP do host ou nome DNS packets - número de pacotes interval - tempo entre pacotes sucessivos em milissegundos size - tamanho do pacote em bytes timeout - tempo limite em milissegundos	Exemplo: => icmpping[,4] → se ao menos um pacote dos quatro for retornado, o item retornará 1. Veja também: tabela de valores padrão .
icmppingloss[<target>,<packets>,<interval>,<size>,<timeout>]	Porcentagem de pacotes perdidos.	Número flutuante.	target - IP do host ou nome DNS packets - número de pacotes interval - tempo entre pacotes sucessivos em milissegundos size - tamanho do pacote em bytes timeout - tempo limite em milissegundos	Veja também: tabela de valores padrão .
icmppingsec[<target>,<packets>,<interval>,<size>,<timeout>,<mode>]				

Chave

tempo de
resposta de
ping ICMP (em
segundos).

Número
flutuante.

target - IP do
host ou nome
DNS

packets -
número de
pacotes

interval -
tempo entre
pacotes
sucessivos em
milissegundos

size - tamanho
do pacote em
bytes

timeout -
tempo limite
em
milissegundos

mode - valores
possíveis: min,
max, avg
(padrão)(mínimo,
máximo e
média)

Pacotes que
são perdidos
ou expirados
não são usados
no cálculo.

Se host não
está disponível
(tempo limite
alcançado), o
item retornará
0.

Se o valor de
retorno for
menor que
0.0001
segundos, o
valor será
fizzado em
0.0001
segundos.

Veja também:
tabela de
valores padrão.

net.tcp.service[service,<ip>,<port>]

Chave

Verifica se o serviço está sendo executado e aceitando conexões TCP.

0 - serviço está parado

1 - serviço está em execução

service - valores possíveis: ssh, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet (veja [detalhes](#))
ip - endereço IP ou nome DNS (por padrão IP/DNS do host é usado)
port - número da porta (por padrão é usado o número de porta padrão do serviço).

Exemplo:
 => net.tcp.service[ftp,,45]
 → pode ser usado para testar disponibilidade de servidor FTP na porta 45/TCP.

Note que com serviço tcp a indicação de porta é obrigatória. Estas verificações podem resultar em mensagens adicionais nos arquivos de log dos daemons de serviço (comumente sessões SMTP e SSH sendo logadas). Verificação de protocolos criptografados (como IMAP na porta 993 ou POP na porta 995) não são suportados atualmente. Como alternativa, por favor utilize net.tcp.service[tcp,<ip>,<port>] para verificações como estas. Os serviços https e telnet são suportados desde o Zabbix 2.0.

net.tcp.service.perf[service,<ip>,<port>]

Chave

	Verifica performance de serviço TCP.	Número flutuante. 0.000000 - serviço está parado seconds - o número de segundos gastos aguardando pela resposta do serviço	service - valores possíveis: ssh, ldap, smtp, ftp, http, pop, nntp, imap, tcp, https, telnet (veja detalhes) ip - endereço IP ou nome DNS (por padrão IP/DNS do host é usado) port - número de porta (por padrão é usado o número de porta padrão do serviço).	Exemplo: => net.tcp.service.perf[ssh] → pode ser usado para testar a velocidade de resposta inicial de um servidor SSH. Note que com serviço tcp a indicação de porta é obrigatória. Verificação de protocolos criptografados (como IMAP na porta 993 ou POP na porta 995) não são suportados atualmente. Como alternativa, por favor utilize net.tcp.service.perf[tcp,<ip>], para verificações como estas. Os serviços https e telnet são suportados desde o Zabbix 2.0. Chamado de tcp_perf antes do Zabbix 2.0.
net.udp.service[service,<ip>,<port>]				

Chave				
	Verifica se o serviço está sendo executado e respondendo a requisições UDP.	0 - serviço está parado 1 - serviço está sendo executado	service - valores possíveis: ntp (veja detalhes) ip - endereço IP ou nome DNS (por padrão o IP/DNS do host é usado) port - número de porta (por padrão é usado o número de porta padrão do serviço).	Exemplo: => net.udp.service[ntp,,45] → pode ser usado para testar a disponibilidade do serviço NTP na porta 45/UDP. Este item é suportado desde o Zabbix 3.0, mas o serviço ntp estava disponível para o item net.tcp.service[] em versões anteriores.
net.udp.service.perf[service,<ip>,<port>]	Verifica a performance de serviço UDP.	Número flutuante. 0.000000 - serviço está parado seconds - o número de segundos gastos aguardando pela resposta do serviço	service - valores possíveis: ntp (veja detalhes) ip - endereço IP ou nome DNS (por padrão IP/DNS do host é usado) port - número de porta (por padrão é usado o número de porta padrão do serviço).	Exemplo: => net.udp.service.perf[ntp] → pode ser usado para testar o tempo de resposta do serviço NTP. Este item é suportado desde o Zabbix 3.0, mas o serviço ntp estava disponível para o item net.tcp.service[] em versões anteriores.

Note that for SourceIP support in LDAP simple checks (since Zabbix 6.0.1), OpenLDAP version 2.6.1 or above is required.

Tempo limite de processamento

O Zabbix não processará uma verificação simples por mais tempo do que os segundos de limite (Timeout) definidos no arquivo de configuração do Zabbix Server/Proxy.

Ping ICMP

O Zabbix usa o utilitário externo **fping** para processar os pings ICMP.

O utilitário não faz parte da distribuição do Zabbix e precisa ser instalado individualmente. Se ele estiver ausente, com permissões insuficientes ou sua localização for diferente do valor definido no arquivo de configuração do Zabbix Server (parâmetro 'FpingLocation'), os itens de ICMP (**icmpping**, **icmppingloss**, **icmppingsec**) não serão processados.

Veja também: [problemas conhecidos](#)

O **fping** precisa ser executável pelos daemons do Zabbix rodando com 'setuid' do root. Execute estes comandos como **root** para configurar corretamente as permissões:

```
shell> chown root:zabbix /usr/sbin/fping
```

```
shell> chmod 4710 /usr/sbin/fping
```

Após executar os dois comandos acima, verifique o proprietário do executável **fping**. Em alguns casos o proprietário pode ser reconfigurado executando o comando `chmod`.

Verifique também se o usuário `zabbix` pertence ao grupo `zabbix` executando:

```
shell> groups zabbix
```

e se não pertencer, adicione-o da seguinte forma:

```
shell> usermod -a -G zabbix zabbix
```

Padrões, limites e descrições dos valores para os parâmetros de verificações ICMP:

Parâmetro	Unidade	Descrição	Fping's flag	Padrões definidos por	Limites permitidos	por
pacotes	number	número de pacotes de solicitação para um destino	-C	3 1 10000	intervalo	milissegundos
tamanho	bytes	tamanho do pacote em bytes	-b	56 ou 68	tempo de espera entre pacotes sucessivos	-p
tempo limite	milissegundos	tempo limite de espera após o último pacote enviado, afetado pelo sinalizador	-C	1000	tempo limite individual para cada pacote	-t

Além disso, o Zabbix usa as opções de `fping -i interval ms` (não confunda com o parâmetro de item intervalo mencionado na tabela acima, que corresponde à opção de `fping -p`) e endereço IP de origem `-S` (ou `-I` em versões mais antigas de `fping`). Essas opções são detectadas automaticamente executando verificações com diferentes combinações de opções. O Zabbix tenta detectar o valor mínimo em milissegundos que o `fping` permite usar com `-i` tentando 3 valores: 0, 1 e 10. O primeiro valor bem-sucedido é então usado para verificações ICMP subsequentes. Esta ação é executada por cada **processo de ping ICMP** individualmente.

As opções de `fping` detectadas automaticamente são invalidadas a cada hora e detectadas novamente na próxima tentativa de realizar a verificação ICMP. Defina `DebugLevel>=4` para visualizar os detalhes desse processo no arquivo de log do Zabbix Server ou Proxy.

Warning:
Aviso: os padrões de `fping` podem diferir dependendo da plataforma e versão - em caso de dúvida, verifique a documentação do `fping`.

O Zabbix grava os endereços IP a serem verificados por qualquer uma das três chaves `icmpping*` em um arquivo temporário, que é então passado para o **fping**. Se os itens têm parâmetros de chave diferentes, apenas aqueles com parâmetros de chave idênticos são gravados em um único arquivo.

Todos os endereços IP gravados no arquivo único serão verificados pelo `fping` em paralelo, então o processo de `ping icmp` do Zabbix levará uma quantidade fixa de tempo, desconsiderando o número de endereços IP no arquivo.

1 Chaves de item de monitoramento VMware

Chaves de item

A tabela fornece detalhes sobre verificações simples que podem ser utilizadas para monitoramento de **ambientes VMware**.

Chave	Descrição	Valor de retorno	Parâmetros	Comentários
vmware.cl.perfcounter[<url>,<id>,<path>,<instance>]	Métricas de contador de performance para cluster VMware.	Inteiro	url - URL de serviço VMware id - ID do cluster VMware path - caminho do contador de performance instance - instância de contador de performance	O <code>id</code> pode ser recebido do <code>vmware.cluster.discovery[]</code> como <code>{#CLUSTER.ID}</code>
vmware.cluster.discovery[<url>]				

Chave

vmware.cluster.status[<url>, <name>]	Descoberta de clusters VMware.	Objeto JSON	url - URL de serviço VMware
	estado do cluster VMwre.	Inteiro: 0 - cinza; 1 - verde; 2 - amarelo; 3 - vermelho	url - URL de serviço VMware name - nome de cluster VMwre
vmware.datastore.discovery[<url>]	Descoberta de datastores VMware.	Objeto JSON	url - URL de serviço VMware
vmware.datastore.hv.list[<url>,<datastore>]	Lista de datastore hypervisors.	Objeto JSON	url - URL de serviço VMware datastore - nome do datastore
vmware.datastore.read[<url>,<datastore>,<mode>]	Quantidade de tempo para uma operação de leitura a partir do datastore (milissegundos).	Inteiro ²	url - URL de serviço VMware datastore - nome do datastore mode - latência (valor médio, padrão), maxlatency (valor máximo)
vmware.datastore.size[<url>,<datastore>,<mode>]	Espaço em bytes ou em porcentagem do total do datastore VMware.	Inteiro - para bytes Número flutuante - para porcentagem	url - URL de serviço VMware datastore - nome do datastore mode - valores possíveis: total (padrão), free, pfree (livre, porcentagem), uncommitted
vmware.datastore.write[<url>,<datastore>,<mode>]	Quantidade de tempo para uma operação de leitura no datastore (milissegundos).	Inteiro ²	url - URL de serviço VMware datastore - nome do datastore mode - latência (valor médio, padrão), maxlatency (valor máximo)

Chave				
vmware.dc.discovery[<url>]	Descoberta de datacenters VMware.	Objeto JSON	url - URL de serviço VMware	
vmware.eventlog[<url>,<mode>]	Log de evento VMware.	Log	url - URL de serviço VMware mode - all (padrão), skip - pula o processamento de dados antigos	Deve haver apenas uma chave de item vmware.eventlog[] por URL. Veja também: exemplo de filtragem de registros de log VMware.
vmware.fullname[<url>]	Nome completo do serviço VMware.	String	url - URL de serviço VMware	
vmware.hv.cluster.name[<url>,<uuid>]	Nome do cluster do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.cpu.usage[<url>,<uuid>]	Uso de processador pelo hypervisor VMware (Hz).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.cpu.usage.perf[<url>,<uuid>]	Uso de processador pelo hypervisor VMware como uma porcentagem durante o intervalo.	Número flutuante	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.cpu.utilization[<url>,<uuid>]	Uso de processador pelo hypervisor VMware como uma porcentagem durante o intervalo, dependente do gerenciamento de energia ou HT.	Número flutuante	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.datacenter.name[<url>,<uuid>]				

Chave			
	Nome do datacenter do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.datastore.discovery[<url>,<uuid>]	Descoberta de datastores de hypervisor VMware.	Objeto JSON	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.datastore.list[<url>,<uuid>]	Lista de datastores de hypervisor VMware.	Objeto JSON	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.datastore.multipath[<url>,<uuid>,<datastore>,<partitionid>]	Número de caminhos de datastores disponíveis.	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware datastore - nome do datastore partitionid - ID interno do dispositivo físico do vmware.hv.datastore.discovery
vmware.hv.datastore.read[<url>,<uuid>,<datastore>,<mode>]	Quantidade de tempo médio para uma operação de leitura a partir do datastore (milissegundos).	Inteiro ²	url - URL de serviço VMware uuid - nome de host do hypervisor VMware datastore - nome do datastore mode - latência (padrão)
vmware.hv.datastore.size[<url>,<uuid>,<datastore>,<mode>]			

	Espaço do datastore VMware em bytes ou em porcentagem do total.	Inteiro - para bytes Número flutuante - para porcentagem	url - URL de serviço VMware uuid - nome de host do hypervisor VMware datastore - nome do datastore mode - valores possíveis: total (padrão), free, pfree (livre, porcentagem), uncommitted
vmware.hv.datastore.write[<url>,<uuid>,<datastore>,<mode>]	Quantidade de tempo médio para uma operação de escrita para o datastore (milissegundos).	Inteiro ²	url - URL de serviço VMware uuid - nome de host do hypervisor VMware datastore - nome do datastore mode - latência (padrão)
vmware.hv.discovery[<url>]	Descoberta de hypervisors VMware.	Objeto JSON	url - URL de serviço VMware
vmware.hv.fullname[<url>,<uuid>]	Nome do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.hw.cpu.freq[<url>,<uuid>]	Frequência de processador do hypervisor VMware (Hz).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.hw.cpu.model[<url>,<uuid>]	Modelo de processador do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.hw.cpu.num[<url>,<uuid>]			

Chave				
	Número de cores de processador no hypervisor VMware.	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.hw.cpu.threads[<url>,<uuid>]	Número de threads de processador no hypervisor VMware.	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.hw.memory[<url>,<uuid>]	Tamanho de memória total do VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.hw.model[<url>,<uuid>]	Modelo de hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.hw.uuid[<url>,<uuid>]	UUID de BIOS do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.hw.vendor[<url>,<uuid>]	Nome do fornecedor do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.maintenance[<url>,<uuid>]	Estado de manutenção do hypervisor VMware.	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	Retorna '0' - não em manutenção ou '1' - em manutenção
vmware.hv.memory.size.ballooned[<url>,<uuid>]				

Chave			
	Tamanho de memória inflada do hypervisor VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.memory.used[<url>,<uuid>]	Tamanho de memória usada pelo hypervisor VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware
vmware.hv.network.in[<url>,<uuid>,<mode>]	Estatísticas de entrada de rede do hypervisor VMware (bytes por segundo).	Inteiro ²	url - URL de serviço VMware uuid - nome de host do hypervisor VMware mode - bps (padrão)
vmware.hv.network.out[<url>,<uuid>,<mode>]	Estatísticas de saída de rede do hypervisor VMware (bytes por segundo).	Inteiro ²	url - URL de serviço VMware uuid - nome de host do hypervisor VMware mode - bps (padrão)
vmware.hv.perfcounter[<url>,<uuid>,<path>,<instance>]	Valor de contador de performance do hypervisor VMware.	Inteiro ²	url - URL de serviço VMware uuid - nome de host do hypervisor VMware path - caminho do contador de performance ¹ instance - instância do contador de performance. Use instância vazia para agragar valores (padrão)
vmware.hv.power[<url>,<uuid>,<max>]			

Chave				
	Uso de energia por hypervisor VMware (W).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware max - máximo uso de energia permitido	
vmware.hv.sensor.health.state[<url>,<uuid>]	Sensor de estado de integridade do hypervisor VMware.	Inteiro: 0 - cinza; 1 - verde; 2 - amarelo; 3 - vermelho	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.sensors.get[<url>,<uuid>]	Sensores de estado de HW de fornecedor do hypervisor VMware.	JSON	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.status[<url>,<uuid>]	Estado do hypervisor VMware.	Inteiro: 0 - cinza; 1 - verde; 2 - amarelo; 3 - vermelho	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	Usa a propriedade de estado geral do sistema do host.
vmware.hv.uptime[<url>,<uuid>]	Tempo ativo do hypervisor VMware (segundos).	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.version[<url>,<uuid>]	Versão do hypervisor VMware.	String	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.hv.vm.num[<url>,<uuid>]	Número de máquinas virtuais no hypervisor VMware.	Inteiro	url - URL de serviço VMware uuid - nome de host do hypervisor VMware	
vmware.version[<url>]	Versão do serviço VMware.	String	url - URL de serviço VMware	

Chave

vmware.vm.cluster.name[<url>,<uuid>]	Nome da máquina virtual VMware.	String	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.cpu.latency[<url>,<uuid>]	Porcentagem de tempo em que a máquina virtual é incapaz de ser executada porque está disputando acesso a CPU física.	Número flutuante	url - URL de serviço VMware uuid - nome de host da máquina virtual no VMware
vmware.vm.cpu.num[<url>,<uuid>]	Número de processadores nas máquinas virtuais no VMware.	Inteiro	url - URL de serviço VMware uuid - nome de máquina virtual do VMware
vmware.vm.cpu.readiness[<url>,<uuid>,<instance>]	Porcentagem de tempo que a máquina virtual está pronta, mas não consegue agendar execução na CPU física.	Número flutuante	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de CPU
vmware.vm.cpu.ready[<url>,<uuid>]	Tempo (em milissegundos) que a máquina virtual está pronta, mas não consegue agendar execução na CPU física. O tempo de CPU ready é dependente do número de máquinas virtuais no host e suas cargas de CPU (%).	Inteiro ²	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.cpu.swapwait[<url>,<uuid>,<instance>]			

Chave

	Porcentagem de tempo de CPU gasto aguardando por swap-in.	Número flutuante	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de CPU
vmware.vm.cpu.usage[<url>,<uuid>]	Uso de processador por máquina virtual VMware (Hz).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.cpu.usage.perf[<url>,<uuid>]	Uso de processador por máquina virtual VMware como uma porcentagem durante o intervalo.	Número flutuante	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.datacenter.name[<url>,<uuid>]	Nome do datacenter de máquinas virtuais VMware.	String	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.discovery[<url>]	Descoberta de máquinas virtuais VMware.	Objeto JSON	url - URL de serviço VMware
vmware.vm.guest.memory.size.swapped[<url>,<uuid>]	Quantidade de memória física de convidado que é enviada como swap para o espaço de swap (KB).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.guest.osuptime[<url>,<uuid>]	Tempo total decorrido desde a última inicialização de sistema operacional (em segundos).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.hv.name[<url>,<uuid>]			

Chave			
	Nome de hypervisor de máquina virtual VMware.	String	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size[<url>,<uuid>]	Tamnhó de memória total de máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.ballooned[<url>,<uuid>]	Tamanho de memória inflada (balloon) da máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.compressed[<url>,<uuid>]	Tamanho de memória comprimida da máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.consumed[<url>,<uuid>]	Quantidade de memória física do host consumida para fazer backup das páginas de memória física de convidado (KB).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.private[<url>,<uuid>]	Tamanho de memória privada da máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.shared[<url>,<uuid>]	Tamanho de memória compartilhada da máquina virtual (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.swapped[<url>,<uuid>]			

Chave

	Tamanho de memória em swap da máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.usage.guest[<url>,<uuid>]	Uso de memória por máquina virtual VMware convidada (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.size.usage.host[<url>,<uuid>]	Uso de memória do host por máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.memory.usage[<url>,<uuid>]	Porcentagem de memória física do host que foi consumida.	Número flutuante	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.net.if.discovery[<url>,<uuid>]	Descoberta de interfaces de rede de máquina virtual VMware.	Objeto JSON	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.net.if.in[<url>,<uuid>,<instance>,<mode>]	Estatísticas de entrada de interface de rede da máquina virtual (bytes/pacotes por segundo).	Inteiro ²	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de interface de rede mode - bps (padrão)/pps - bytes/pacotes por segundo
vmware.vm.net.if.out[<url>,<uuid>,<instance>,<mode>]			

	Estatísticas de saída de interface de rede da máquina virtual VMware (bytes/pacotes por segundo).	Inteiro ²	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de interface de rede mode - bps (padrão)/pps - bytes/pacotes por segundo
vmware.vm.net.if.usage[<url>,<uuid>,<instance>]	Utilização de rede da máquina virtual VMware (combinadas taxa de transmissão e taxa de recepção (transmit-rates e receive-rates)) durante o intervalo (KBps).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de interface de rede
vmware.vm.perfcounter[<url>,<uuid>,<path>,<instance>]	Valor de contador de performance da máquina virtual VMware.	Inteiro ²	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware path - caminho do contador de performance ¹ instance - instância do contador de performance. Use instância vazia para agregar valores (padrão)
vmware.vm.powerstate[<url>,<uuid>]	Estado de energia da máquina virtual VMware.	Inteiro: 0 - desligada; 1 - ligada; 2 - suspesa	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.storage.committed[<url>,<uuid>]			

Chave

	Espaço de armazenamento entregue à máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.storage.readio[<url>,<uuid>,<instance>]	Número médio de solicitações de leitura pendentes para o disco virtual durante o intervalo de coleta.	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de dispositivo de disco (obrigatório)
vmware.vm.storage.totalreadlatency[<url>,<uuid>,<instance>]	O tempo médio que uma leitura a partir dos discos virtuais leva (milissegundos).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de dispositivo de disco (obrigatório)
vmware.vm.storage.totalwritelatency[<url>,<uuid>,<instance>]	O tempo médio que uma escrita nos discos virtuais leva (milissegundos).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware instance - instância de dispositivo de disco (obrigatório)
vmware.vm.storage.uncommitted[<url>,<uuid>]	Espaço de armazenamento revogado da máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware
vmware.vm.storage.unshared[<url>,<uuid>]	Espaço de armazenamento descompartilhado na máquina virtual VMware (bytes).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware

Chave

vmware.vm.storage.writeoio[<url>,<uuid>,<instance>]

Número médio de solicitações de escrita pendente para o disco virtual durante o intervalo de coleta.

Inteiro

url - URL de serviço VMware
uuid - nome de host de máquina virtual VMware
instance - instância de dispositivo de disco (obrigatório)

vmware.vm.uptime[<url>,<uuid>]

Tempo ativo da máquina virtual VMware (segundos).

Inteiro

url - URL de serviço VMware
uuid - nome de host de máquina virtual VMware

vmware.vm.vfs.dev.discovery[<url>,<uuid>]

Descoberta de dispositivos de disco de máquina virtual VMware.

Objeto JSON

url - URL de serviço VMware
uuid - nome de host de máquina virtual VMware

vmware.vm.vfs.dev.read[<url>,<uuid>,<instance>,<mode>]

Estatísticas de leitura de dispositivos de disco da máquina virtual VMware (bytes/operações por segundo).

Inteiro ²

url - URL de serviço VMware
uuid - nome de host de máquina virtual VMware
instance - instância de dispositivo de disco
mode - bps (padrão)/ops - bytes/operações por segundo

vmware.vm.vfs.dev.write[<url>,<uuid>,<instance>,<mode>]

Estatísticas de escrita dos dispositivos de disco da máquina virtual VMware (bytes/operações por segundo).

Inteiro ²

url - URL de serviço VMware
uuid - nome de host de máquina virtual VMware
instance - instância de dispositivo de disco
mode - bps (padrão)/ops - bytes/operações por segundo

Chave				
vmware.vm.vfs.fs.discovery[<url>,<uuid>]	Descoberta de sistemas de arquivo de máquina virtual VMware.	Objeto JSON	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware	O VMware Tools deve estar instalado nas máquinas virtuais (guest).
vmware.vm.vfs.fs.size[<url>,<uuid>,<fsname>,<mode>]	Estatísticas de sistema de arquivo da máquina virtual VMware (bytes/porcentagens).	Inteiro	url - URL de serviço VMware uuid - nome de host de máquina virtual VMware fsname - nome do sistema de arquivo mode - total/free/used/pfree/pused	O VMware Tools deve estar instalado nas máquinas virtuais (guest).

Notas de rodapé

¹ O caminho de contador de performance do VMware tem o formato `group/counter[rollup]` onde:

- `group` - o grupo de contador do performance, por exemplo `cpu`
- `counter` - o nome do contador de performance, por exemplo `usagemhz`
- `rollup` - o tipo de rollup de contador de performance, por exemplo `average`

Assim o exemplo acima daria o seguinte caminho de contador: `cpu/usagemhz[average]`

As descrições de grupo de contador de performance, nomes de contador e tipos de rollup podem ser encontrados na [documentação do VMware](#).

² O valor destes itens é obtido dos contadores de performance do VMware e o **parâmetro** `VMwarePerfFrequency` é usado para atualizar seus dados no cache de VMware do Zabbix:

- `vmware.hv.datastore.read`
- `vmware.hv.datastore.write`
- `vmware.hv.network.in`
- `vmware.hv.network.out`
- `vmware.hv.perfcounter`
- `vmware.vm.cpu.ready`
- `vmware.vm.net.if.in`
- `vmware.vm.net.if.out`
- `vmware.vm.perfcounter`
- `vmware.vm.vfs.dev.read`
- `vmware.vm.vfs.dev.write`

Mais informações

Consulte **monitoramento de máquinas virtuais** para informações detalhadas em como configurar o Zabbix para monitorar ambientes VMware.

6 Monitoramento de arquivo de log

Visão geral

O Zabbix pode ser usado para análise e monitoramento de arquivos de log com/sem suporte a rotação de log.

Notificações podem ser usadas para alertar os usuários quando um arquivo de log contiver alguma frase ou padrão de texto.

Para monitorar um arquivo de log você deve ter:

- um Zabbix Agent sendo executado no host
- um item de monitoramento de log configurado

Attention:

O tamanho de um arquivo de log monitorado depende do [suporte a arquivos grandes](#).

Configuração

Verifique os parâmetros do agente

Certifique-se de que no [arquivo de configuração do agente](#):

- O parâmetro 'Hostname' corresponde ao nome do host no Zabbix Frontend (interface web)
- Os Zabbix Servers no parâmetro 'ServerActive' estão definidos para o processamento de verificações ativas

Configuração do item

Configure um [item](#) de monitoramento de log.

Item	Tags	Preprocessing						
* Name	Log item							
Type	Zabbix agent (active) ▼							
* Key	log[/var/log/syslog,error]							
Type of information	Log ▼							
* Update interval	30s							
Custom intervals	<table border="1"> <thead> <tr> <th>Type</th> <th>Interval</th> <th>Period</th> </tr> </thead> <tbody> <tr> <td colspan="3">Add</td> </tr> </tbody> </table>		Type	Interval	Period	Add		
Type	Interval	Period						
Add								
* History storage period	<div>Do not keep history</div> <div>Storage period 3600</div>							
Log time format	ppppddphh:mm:ss							

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Especificamente para itens de monitoramento de log você informa:

Tipo

Chave

Selecione **Zabbix Agent (ativo)** aqui.

Use uma das seguintes chaves de item:

log[] ou **logrt[]**:

Estas duas chaves de item permitem monitorar logs e filtrar entradas de log pelo conteúdo da expressão regular, se houver.

Por exemplo: log[/var/log/syslog,error].

Certifique-se de que o arquivo permite leitura pelo usuário 'zabbix', caso contrário o estado do item se tornará 'não suportado'.

log.count[] ou **logrt.count[]**:

Estas duas chaves de item permitem retornar apenas o número de linhas correspondentes.

Consulte a seção [chave de item suportado pelo Zabbix Agent](#) para detalhes de utilização destas chaves e seus parâmetros.

Tipo de informação	<p>Preenchido automaticamente:</p> <p>Para itens log[] ou logrt[] - Log;</p> <p>Para itens log.count[] ou logrt.count[] - Numérico (unsigned).</p> <p>Se estiver usando o parâmetro opcional output, você pode selecionar manualmente o tipo de informação apropriado diferente de Log.</p> <p>Note que a escolha de um tipo de informação não-log acarretará na perda do registro de data e hora local.</p>
Intervalo de atualização (em seg)	<p>O parâmetro define quão frequentemente o Zabbix Agent verificará por qualquer alteração no arquivo de log.</p> <p>Configurar este parâmetro para 1 segundo garantirá que você obtenha novos registros tão logo quanto possível.</p>
Formato de data e hora do log	<p>Neste campo você pode opcionalmente especificar o padrão para analisar o registro de data e hora do arquivo de log.</p> <p>Se deixado em branco o registro de data e hora não será analisado.</p> <p>Marcadores de posição suportados:</p> <ul style="list-style-type: none"> * y: Ano (0001-9999) * M: Mês (01-12) * d: Dia (01-31) * h: Hora (00-23) * m: Minuto (00-59) * s: Segundo (00-59) <p>Por exemplo, considere a seguinte linha do arquivo de log do Zabbix Agent:</p> <p>" 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211)."</p> <p>Ela começa com seis posições de caracter para o PID, seguido da data, hora, e o restante da linha.</p> <p>O formato de data e hora para esta linha seria "pppppp:yyyyMMdd:hhmmss".</p> <p>Note que os caracteres "p" e ":" são apenas marcadores de posição e podem ser qualquer coisa, exceto "yMdhms".</p>

Notas importantes

- O Zabbix Server e Agent mantêm o registro do tamanho e data da última modificação (para logrt) do arquivo de log em dois contadores. Adicionalmente:
 - O agente também usa internamente números de inode (no UNIX/GNU/Linux), índices de arquivos (no Microsoft Windows) e soma de verificação MD5 dos primeiros 512 bytes do arquivo de log para melhorar a detecção de arquivos de log truncados e rotacionados.
 - Nos sistemas UNIX/GNU/Linux é assumido que os sistemas de arquivo onde os arquivos de log são armazenados reportem números de inode, que podem ser usados para rastrear arquivos.
 - No Microsoft Windows o Zabbix Agent determina o tipo de sistema de arquivo no qual os arquivos de log residem e usa:
 - * Nos sistemas de arquivo NTFS índices de arquivo 64-bit.
 - * Nos sistemas de arquivo ReFS (apenas de Microsoft Windows Server 2012) IDs de arquivo 128-bit.
 - * Nos sistemas de arquivo onde índices de arquivo mudam (p.e. FAT32, exFAT) um algoritmo alternativo é usado para uma abordagem sensível em condições incertas quando a rotação de log resulta em múltiplos arquivos de log com a mesma data e hora de modificação.
 - Os números de inode, índices de arquivo e somas de verificação MD5 são internamente coletados pelo Zabbix Agent. Eles não são transmitidos para o Zabbix Server e são perdidos quando o Zabbix Agent é parado.
 - Não modifique a última data de alteração dos arquivos de log com o utilitário 'touch', não copie um arquivo de log com restauração anterior do arquivo original (isto mudará o número de inode do arquivo). Em ambos os casos o arquivo será considerado diferente e será analisado desde o início, o que pode resultar em alertas duplicados.
 - Se houver vários arquivos de log correspondentes ao item logrt[] e o Zabbix Agent estiver acompanhando o mais recente deles e este arquivo de log mais recente for eliminado, uma mensagem de alerta "não há arquivos correspondentes à "<máscara regexp>" no "<diretório>" será mostrada. O Zabbix Agent ignora arquivos de log com data de modificação menor que a data de modificação mais recente vista pelo agente para o item logrt[] sendo verificado.
- O agente começa lendo o arquivo de log a partir do ponto em que parou na última vez.

- A quantidade de bytes já analisados (o contador de tamanho) e a última data de modificação (o contador de data) são armazenados no banco de dados do Zabbix e são enviados para o agente para garantir que o agente inicie a leitura do arquivo de log deste ponto nos casos quando o agente é recém iniciado ou recebeu itens que foram previamente desabilitados ou não suportados. No entanto, se o agente recebeu um contador de tamanho diferente de zero do servidor, mas o item `logrt[]` ou `logrt.count[]` não foi encontrado e não há arquivos correspondentes, o contador de tamanho é reiniciado para 0 de modo a analisar desde o início no caso de os arquivos aparecerem mais tarde.
- Sempre que o arquivo de log tornar-se menor que o contador de tamanho de log conhecido pelo agente, o contador é reconfigurado para zero e o agente começa a leitura do arquivo de log do começo, levando em conta o contador de data.
- Se houver vários arquivos correspondentes com a mesma data de última alteração dentro do diretório, então o agente tenta analisar corretamente todos os arquivos de log com a mesma data de modificação evitando pular dados ou analisar dados repetidos, apesar deste comportamento não poder ser garantido em todas as situações. O agente não pressupõe qualquer esquema de rotação de arquivo de log e nem determina um. Quando forem apresentados múltiplos arquivos de log com a mesma data de última alteração, o agente os processará em uma ordem lexicográfica decrescente. Assim, para alguns esquemas de rotação os arquivos de log serão analisados e reportados em sua ordem original. Para outros esquemas de rotação a ordem original dos arquivos de log não será honrada, o que pode acarretar na entrega de registros de arquivos de log em ordem alterada (o problema não ocorre se os arquivos de log tiverem datas de última alteração diferentes).
- O Zabbix Agent processa novos registros de um arquivo de log uma vez a cada Intervalo de atualização segundos.
- O Zabbix Agent não envia mais do que **maxlines** de um arquivo de log por segundo. O limite evita sobrecarga de recursos de rede e CPU e sobrescreve o valor padrão fornecido pelo parâmetro **MaxLinesPerSecond** no [arquivo de configuração do agente](#).
- Para encontrar o texto requerido o Zabbix irá processar 10 vezes mais novas linhas do que configurado em **MaxLinesPerSecond**. Assim, por exemplo, se um item `log[]` ou `logrt[]` possui Intervalo de atualização de 1 segundo, por padrão o agente analisará não mais do que 200 registros do arquivo de log e enviará não mais do que 20 registros correspondentes para o Zabbix Server em uma verificação. Aumentando **MaxLinesPerSecond** no arquivo de configuração do agente ou configurando o parâmetro **maxlines** na chave do item, o limite pode ser elevado até 10000 registros de log analisados e 1000 registros compatíveis enviados para o Zabbix Server em uma verificação. Se o Intervalo de atualização estiver configurado para 2 segundos então os limites para uma verificação seriam configurados 2 vezes mais altos do que com Intervalo de atualização de 1 segundo.
- Adicionalmente, os valores `log` e `log.count` são sempre limitados a 50% do tamanho de buffer de envio do agente, mesmo que não haja valores não-log configurados. Então para os valores **maxlines** serem enviados em uma conexão (e não em várias conexões), o parâmetro de [tamanho de buffer \(BufferSize\)](#) do agente deve ser de pelo menos **maxlines** x 2.
- Na ausência de itens de log todo o tamanho de buffer do agente é utilizado para valores não-log. Quando vierem valores de log estes substituirão os valores não-log mais antigos conforme necessário, até o limite de 50%.
- Para registros de arquivo de log maiores que 256kB, apenas os primeiros 256kB são analisados com a expressão regular e o restante dos registros é ignorado. No entanto, se o Zabbix Agent for parado enquanto estiver lidando com um registro longo, o estado interno do agente é perdido e o registro longo pode ser analisado novamente e de forma diferente assim que o agente for reiniciado.
- Nota especial para separadores de caminho `"\"`: se `file_format` (formato do arquivo) for `"file\log"`, então não deve existir um diretório `"file"`, desde que não é possível de forma não ambígua definir se `"."` está escapado ou se é o primeiro símbolo do nome do arquivo.
- Expressões regulares para `logrt` são suportadas apenas no nome do arquivo; expressão regular para correspondência de diretório não é suportada.
- Nas plataformas UNIX um item `logrt[]` se torna NÃO SUPORTADO (NOTSUPPORTED) se um diretório onde se espera encontrar os arquivos de log não existir.
- No Microsoft Windows, se um diretório não existe o item não se tornará NÃO SUPORTADO (por exemplo, se o diretório contiver erro de digitação na chave do item).
- A ausência de arquivos de log para o item `logrt[]` não o torna NÃO SUPORTADO. Erros de leitura de arquivos de log para o item `logrt[]` são registrados como alertas dentro do arquivo de log do Zabbix Agent mas não tornam o item NÃO SUPORTADO.
- O arquivo de log do Zabbix Agent pode ser útil para encontrar o por quê de um item `log[]` ou `logrt[]` tornar-se NÃO SUPORTADO. O Zabbix pode monitorar seu arquivo de log, com exceção de quando estiver em `DebugLevel=4`.

Extraindo a parte correspondente de uma expressão regular

Algumas vezes podemos querer extrair apenas um valor de interesse de um arquivo alvo, em vez de retornar a linha completa de uma correspondência à uma expressão regular.

Desde o Zabbix 2.2.0, itens de log têm a habilidade de extrair valores desejados de linhas correspondentes. Isto é possível com o parâmetro adicional **output** nos itens `log` e `logrt`.

O uso do parâmetro 'output' permite indicar o "grupo de captura" na correspondência em que possamos estar interessados.

Então, por exemplo

```
.log[/path/to/the/file,"large result buffer allocation.*Entries: ([0-9]+)","",\1]
```

deve permitir retornar a contagem de entradas (entries) de acordo com o encontrado no conteúdo de::

```
·Fr Feb 07 2014 11:07:36.6690 */ Thread Id 1400 (GLEWF) large result ·buffer allocation - /Length: 437136/Entries: 5948/Client Ver:
>=10/RPC ·ID: 41726453/User: AUser/Form: CFG:ServiceLevelAgreement
```

Apenas o número será retornado porque \1 se refere apenas ao primeiro e único grupo de captura: ([0-9]+).

E, com a habilidade de extrair e retornar um número, o valor pode ser usado para definir gatilhos.

Notas sobre rotação de arquivo de log com 'copytruncate'

O logrt com a opção copytruncate assume que arquivos de log diferentes possuem registros diferentes (ao menos seus registros de data são diferentes), portanto a soma de verificação MD5 dos blocos iniciais (até os primeiros 512 bytes) será diferente. Dois arquivos com a mesma soma de verificação MD5 implica que um deles é o original, e outro uma cópia.

O logrt com a opção copytruncate se esforça para processar corretamente cópias de arquivo de log sem reportar duplicatas. No entanto, situações como a produção de vários arquivos de log com o mesmo registro de data, rotação de arquivo de log mais frequente do que o intervalo de atualização do item logrt[], e reinício frequente do agente não são recomendados. O agente procura lidar com todas estas situações de forma razoavelmente bem, mas bons resultados não podem ser garantidos em todas as circunstâncias.

Notas sobre arquivos persistentes para itens log*[]

Propósito dos arquivos persistentes

Quando o Zabbix Agent é iniciado ele recebe uma lista de verificações ativas do Zabbix Server ou Proxy. Para métricas de log*[] ele recebe o tamanho do log processado e a data de modificação para encontrar o ponto de partida para monitorar o arquivo de log. Dependendo do tamanho real do arquivo de log e data de modificação informada pelo sistema de arquivos o agente decide em continuar o monitoramento do log a partir tamanho de log processado ou re-analisar o arquivo de log desde o início.

Um agente em execução mantém um extenso conjunto de atributos para o rastreamento de todos os arquivos de log monitorados entre as verificações. Este estado em-memória é perdido quando o agente é parado.

O novo parâmetro opcional **persistent_dir** especifica o diretório para o armazenamento deste estado dos itens de log[], log.count[], logrt[] ou logrt.count[] em um arquivo. O estado de um item de log é restaurado do arquivo persistente após o Zabbix Agent ser reiniciado.

O caso de uso primário é o monitoramento de arquivo de log localizado em um sistema de arquivo espelhado. Até algum momento em que o arquivo de log é gravado em ambos os locais. Então os ambientes espelhados são separados. Na cópia ativa o arquivo de log continua crescendo, recebendo novos registros. O Zabbix Agent o analisa e envia os tamanhos de logs processados e data de modificação para o Server. Na cópia passiva o arquivo de log continua o mesmo, bem atrás da cópia ativa. Mais tarde o sistema operacional e o Zabbix Agent da cópia passiva são reiniciados. O tamanho de log processado e a data de modificação que o Agent recebe do Server pode não ser válida para a situação na cópia passiva. Para continuar o monitoramento do log a partir do lugar em que o agente parou no momento em que o espelhamento do sistema de arquivo foi separado, o agente restaura seu estado de um arquivo persistente.

Operação do agente com arquivo persistente

Na inicialização o Zabbix Agent não sabe nada sobre arquivos persistentes. Apenas após receber uma lista de verificações ativas do Zabbix Server (proxy) o agente identifica que alguns itens de log devem ser apoiados por arquivos persistentes dentro de diretórios especificados.

Durante a operação do agente os arquivos persistentes são abertos para gravação (com fopen(filename, "w")) e sobrescritos com os últimos dados. A chance de perder dados de arquivo persistente se a sobrescrita e a separação dos espelhos de sistema de arquivo ocorrerem ao mesmo tempo é muito pequena, portanto sem manipulação especial para este caso. A gravação em arquivo persistente NÃO é seguida por sincronização forçada de mídia de armazenamento (fsync() não é chamado).

A sobrescrita com os dados mais recentes é feita após o reporte bem-sucedido de registros de arquivo de log ou metadados correspondentes (tamanho de log processado e data de modificação) para o Zabbix Server. Isto pode ocorrer tão frequentemente quanto cada verificação de item se o arquivo de log continuar sendo alterado.

Após receber uma lista de verificações ativas o agente escaneia o diretório de arquivo persistente e remove arquivos persistentes obsoletos. A remoção é efetuado com um atraso de 24h porque arquivos de log em estado NÃO SUPORTADO não são incluídos na lista de verificações ativas, mas eles podem se tornar SUPORTADOS mais tarde e seus arquivos persistentes podem vir a ser úteis. Se o agente for parado e reiniciado antes que as 24h expirem, então os arquivos obsoletos não serão apagados pois o Zabbix Agent não está mais recebendo do Zabbix Server informações sobre suas localizações.

Sem ações especiais durante desligamento do agente.

Nomenclatura e localização de arquivos persistentes

O Zabbix Agent distingue as verificações ativas pelas suas chaves. Por exemplo, logrt[/home/zabbix/test.log] e logrt[/home/zabbix/test.log,] são itens diferentes. Modificar o item logrt[/home/zabbix/test.log,,,10] no Zabbix Frontend para logrt[/home/zabbix/test.log,,,20]

resultará na eliminação do item `logrt[/home/zabbix/test.log,,,10]` da lista de verificações ativas do agente e criará o item `logrt[/home/zabbix/test.log,,,20]` (alguns atributos são mantidos entre modificações no Frontend/Server, mas não no agente).

O nome do arquivo é composto da soma de verificação MD5 da chave do item junto com o seu comprimento para reduzir a possibilidade de colisões. Por exemplo, o estado do item `logrt[/home/zabbix50/test.log,,,,,,/home/zabbix50/agent_private]` será mantido no arquivo persistente `c963ade4008054813bbc0a650bb8e09266`.

Múltiplos itens de log podem usar o mesmo valor de **persistent_dir**.

O parâmetro **persistent_dir** é especificado levando em conta os leiautes do sistema de arquivo específico, pontos de montagem e configuração de espelhamento de armazenamento - o arquivo persistente deve estar no mesmo sistema de arquivo espelhado que o arquivo de log monitorado.

Se o diretório **persistent_dir** não puder ser criado ou não existir, ou os direitos de acesso para o Zabbix Agent não permitirem criar/gravar/ler/apagar arquivos o item de log se torna NÃO SUPORTADO.

Carga de I/O

O arquivo persistente do item é atualizado após o envio bem-sucedido de cada lote de dados (contendo dados do item) para o Server. Por exemplo, o tamanho de 'BufferSize' é 100. Se um item de log encontrou 70 registros correspondentes, então os primeiros 50 registros serão enviados em um lote, o arquivo persistente será atualizado, e então os 20 registros restantes serão enviados (talvez com algum atraso quando houver mais dado acumulado) no segundo lote, e o arquivo persistente será novamente atualizado.

Notas sobre arquivos persistentes para itens `log*[]`

Carga de I/O

O arquivo persistente do item é atualizado após o envio bem-sucedido de cada lote de dados (contendo dados do item) para o Server. Por exemplo, o tamanho de 'BufferSize' é 100. Se um item de log encontrou 70 registros correspondentes, então os primeiros 50 registros serão enviados em um lote, o arquivo persistente será atualizado, e então os 20 registros restantes serão enviados (talvez com algum atraso quando houver mais dado acumulado) no segundo lote, e o arquivo persistente será novamente atualizado.

Ações caso a comunicação entre Agent e Server falhe

Cada linha correspondente dos itens `log[]` e `logrt[]` e um resultado de cada verificação de item `log.count[]` e `logrt.count[]` requerem um slot livre na área designada de 50% no buffer de envio do agente. Os elementos do buffer são regularmente enviados para o Server (ou Proxy) e os slots do buffer ficam livres novamente.

Enquanto houver slots livres na área de log designada no buffer de envio do agente e a comunicação falhar entre Agent e Server (ou Proxy), os resultados de monitoramento de log são acumulados no buffer de envio. Isto ajuda a mitigar falhas de comunicação curtas.

Durante longas falhas de comunicação todos os slots de log ficam ocupados e as seguintes ações são tomadas:

- itens de verificação `log[]` e `logrt[]` são parados. Quando a comunicação é restaurada e slots livres estão disponíveis no buffer as verificações são resumidas da posição anterior. Nenhuma linha correspondente é perdida, elas são apenas registradas mais tarde.
- verificações `log.count[]` e `logrt.count[]` são paradas se `maxdelay = 0` (padrão). O comportamento é similar aos itens `log[]` e `logrt[]` como descrito acima. Note que isto pode afetar os resultados de `log.count[]` e `logrt.count[]`: por exemplo, uma verificação conta 100 linhas correspondentes em um arquivo de log, mas como não há slots livres no buffer a verificação é parada. Quando a comunicação é restabelecida o agente conta as mesmas 100 linhas e mais 70 novas linhas correspondentes. O agente agora envia `count = 170` como se tivesse sido encontrado em uma única verificação.
- verificações `log.count[]` e `logrt.count[]` com `maxdelay > 0`: se não houve "salto" durante a verificação, então o comportamento é similar ao descrito acima. Se um "salto" sobre as linhas do arquivo de log ocorreu então a posição após o "salto" é mantida e o resultado contado é descartado. Assim, o agente tenta acompanhar um arquivo de log em crescimento mesmo em caso de falha de comunicação.

Handling of regular expression compilation and runtime errors

If a regular expression used in `log[]`, `logrt[]`, `log.count[]` or `logrt.count[]` item cannot be compiled by PCRE or PCRE2 library then the item goes into NOTSUPPORTED state with an error message. To continue monitoring the log item, the regular expression should be fixed.

If the regular expression compiles successfully, but fails at runtime (on some or on all log records), then the log item remains supported and monitoring continues. The runtime error is logged in the Zabbix agent log file (without the log file record).

Note that the logging of regular expression runtime errors is supported since Zabbix 6.0.21.

The logging rate is limited to one runtime error per check to allow Zabbix agent to monitor its own log file. For example, if 10 records are analyzed and 3 records fail with a regexp runtime error, one record is produced in the agent log.

Exception: if MaxLinesPerSecond=1 and update interval=1 (only 1 record is allowed to analyze per check) then regexp runtime errors are not logged.

zabbix_agentd logs the item key in case of a runtime error, zabbix_agent2 logs the item ID to help identify which log item has runtime errors. It is recommended to redesign the regular expression in case of runtime errors.

Propósito dos arquivos persistentes

Quando o Zabbix Agent é iniciado ele recebe uma lista de verificações ativas do Zabbix Server ou Proxy. Para métricas de log*[] ele recebe o tamanho do log processado e a data de modificação para encontrar o ponto de partida para monitorar o arquivo de log. Dependendo do tamanho real do arquivo de log e data de modificação informada pelo sistema de arquivos o agente decide em continuar o monitoramento do log a partir tamanho de log processado ou re-analisar o arquivo de log desde o início.

Um agente em execução mantém um extenso conjunto de atributos para o rastreamento de todos os arquivos de log monitorados entre as verificações. Este estado em-memória é perdido quando o agente é parado.

O novo parâmetro opcional **persistent_dir** especifica o diretório para o armazenamento deste estado dos itens de log[], log.count[], logrt[] ou logrt.count[] em um arquivo. O estado de um item de log é restaurado do arquivo persistente após o Zabbix Agent ser reiniciado.

O caso de uso primário é o monitoramento de arquivo de log localizado em um sistema de arquivo espelhado. Até algum momento em que o arquivo de log é gravado em ambos os locais. Então os ambientes espelhados são separados. Na cópia ativa o arquivo de log continua crescendo, recebendo novos registros. O Zabbix Agent o analisa e envia os tamanhos de logs processados e data de modificação para o Server. Na cópia passiva o arquivo de log continua o mesmo, bem atrás da cópia ativa. Mais tarde o sistema operacional e o Zabbix Agent da cópia passiva são reiniciados. O tamanho de log processado e a data de modificação que o Agent recebe do Server pode não ser válida para a situação na cópia passiva. Para continuar o monitoramento do log a partir do lugar em que o agente parou no momento em que o espelhamento do sistema de arquivo foi separado, o agente restaura seu estado de um arquivo persistente.

Operação do agente com arquivo persistente

Na inicialização o Zabbix Agent não sabe nada sobre arquivos persistentes. Apenas após receber uma lista de verificações ativas do Zabbix Server (proxy) o agente identifica que alguns itens de log devem ser apoiados por arquivos persistentes dentro de diretórios especificados.

Durante a operação do agente os arquivos persistentes são abertos para gravação (com fopen(filename, "w")) e sobrescritos com os últimos dados. A chance de perder dados de arquivo persistente se a sobrescrita e a separação dos espelhos de sistema de arquivo ocorrerem ao mesmo tempo é muito pequena, portanto sem manipulação especial para este caso. A gravação em arquivo persistente NÃO é seguida por sincronização forçada de mídia de armazenamento (fsync() não é chamado).

A sobrescrita com os dados mais recentes é feita após o reporte bem-sucedido de registros de arquivo de log ou metadados correspondentes (tamanho de log processado e data de modificação) para o Zabbix Server. Isto pode ocorrer tão frequentemente quanto cada verificação de item se o arquivo de log continuar sendo alterado.

Sem ações especiais durante o desligamento do agente.

Após receber uma lista de verificações ativas o agente marca os arquivos persistentes obsoletos para remoção. Um arquivo persistente se torna obsoleto se: 1) o item de log correspondente não é mais monitorado, 2) um item de log é reconfigurado com uma localização de **persistent_dir** diferente da anterior.

A remoção é efetuado com um atraso de 24h porque arquivos de log em estado NÃO SUPORTADO não são incluídos na lista de verificações ativas, mas eles podem se tornar SUPORTADOS mais tarde e seus arquivos persistentes podem vir a ser úteis.

Se o agente for parado e reiniciado antes que as 24h expirem, então os arquivos obsoletos não serão apagados pois o Zabbix Agent não está mais recebendo do Zabbix Server informações sobre suas localizações.

Warning:

Reconfigurar o **persistent_dir** de um item de log de volta para a localização **persistent_dir** anterior enquanto o agente estiver parado, sem remover o arquivo persistente antigo por usuário - ocasionará a restauração do estado do agente a partir do arquivo persistente antigo, resultando em mensagens perdidas ou falsos alertas.

Nomenclatura e localização de arquivos persistentes

O Zabbix Agent distingue as verificações ativas pelas suas chaves. Por exemplo, logrt[/home/zabbix/test.log] e logrt[/home/zabbix/test.log,] são itens diferentes. Modificar o item logrt[/home/zabbix/test.log,,10] no Zabbix Frontend para logrt[/home/zabbix/test.log,,20] resultará na eliminação do item logrt[/home/zabbix/test.log,,10] da lista de verificações ativas do agente e criará o item logrt[/home/zabbix/test.log,,20] (alguns atributos são mantidos entre modificações no Frontend/Server, mas não no agente).

O nome do arquivo é composto da soma de verificação MD5 da chave do item junto com o seu comprimento para reduzir a possibilidade de colisões. Por exemplo, o estado do item logrt[/home/zabbix50/test.log,,,,,/home/zabbix50/agent_private] será mantido no arquivo persistente c963ade4008054813bbc0a650bb8e09266.

Múltiplos itens de log podem usar o mesmo valor de **persistent_dir**.

O parâmetro **persistent_dir** é especificado levando em conta os leiautes do sistema de arquivo específico, pontos de montagem e configuração de espelhamento de armazenamento - o arquivo persistente deve estar no mesmo sistema de arquivo espelhado que o arquivo de log monitorado.

Se o diretório **persistent_dir** não puder ser criado ou não existir, ou os direitos de acesso para o Zabbix Agent não permitirem criar/gravar/ler/apagar arquivos o item de log se torna NÃO SUPORTADO.

Se os direitos de acesso ao armazenamento de arquivos persistentes forem removidos durante a operação do agente ou outros erros ocorrerem (p.e. disco cheio) então os erros são registrados no arquivo de log do agente, mas o item de log não se torna NÃO SUPORTADO.

7 Itens calculados

Visão geral

Com itens calculados é possível criar cálculos baseados nos valores de outros itens.

Cálculos podem usar ambos:

- valores únicos de itens individuais
- filtros complexos para selecionar múltiplos itens para agregação (consulte [cálculos agregados](#) para detalhes)

Assim, itens calculados são uma forma de criar fontes de dados virtuais. Todos os cálculos são efetuados unicamente pelo Zabbix Server. Os valores são periodicamente calculados com base na expressão aritmética utilizada.

O dado resultante é armazenado no banco de dados do Zabbix assim como para qualquer outro item; ambos os valores históricos e de tendência são armazenados de modo que gráficos podem ser gerados.

Note:

Se o resultado do cálculo for um valor numérico flutuante (float) ele será aparado para um inteiro se o tipo de informação do item calculado for Numérico (unsigned).

Itens calculados compartilham sua sintaxe com [expressões de gatilho](#). A comparação com strings é permitida em itens calculados. Itens calculados podem ser referenciados por macros ou outras entidades assim como qualquer outro tipo de item.

Para usar itens calculados, escolha o tipo de item **Calculado**.

Campos configuráveis

A **chave** é um identificador de item único (por host). Você pode criar qualquer nome de chave usando símbolos suportados.

Definições de cálculos devem ser inseridas no campo **Fórmula**. Não há conexão virtual entre a fórmula e a chave. Os parâmetros de chave não são usados na fórmula de nenhuma forma.

A sintaxe de uma fórmula simples é:

```
function(/host/key,<parameter1>,<parameter2>,...)
```

onde:

função	Uma das funções suportadas : last, min, max, avg, count, etc
host	Host do item usado para o cálculo. O host atual pode ser omitido (p.e. como em <code>function(/key,parameter,...)</code>).
chave	Chave do item usado para o cálculo.
parâmetro(s)	Parâmetros da função, se exigido.

Attention:

Macros de usuário na fórmula serão expandidas se usadas para referenciar um parâmetro de função ou uma constante. Macros de usuário NÃO serão expandidas se referenciando uma função, nome de host, chave de item, parâmetro ou operador de chave de item.

Um fórmula mais complexa pode usar uma combinação de funções, operadores e parêntesis. Você pode usar todas as funções e [operadores](#) suportados nas expressões de gatilho. A lógica e precedência de operador são exatamente as mesmas.

Diferente das expressões de gatilho, o Zabbix processa itens calculados de acordo com o intervalo de atualização do item, não ao receber um novo valor.

Todos os itens que são referenciados pelas funções históricas na fórmula do item calculado devem existir e estar coletando dados. Ademais, se você alterar a chave de um item referenciado, você tem que atualizar manualmente quaisquer fórmulas que façam uso dessa chave.

Um item calculado pode se tornar não suportado em vários casos:

- item(ns) referenciado(s)
 - não foi encontrado
 - está desabilitado
 - pertence a um host desabilitado
 - não é suportado (exceto com função `nodata()` e **operadores** com valores desconhecidos)
- nenhum dado para calcular a função
- divisão por zero
- sintaxe utilizada incorreta

Exemplos de uso

Exemplo 1

Cálculo de porcentagem de espaço livre em disco em '/'.

Uso da função **last**:

```
100*last(/vfs.fs.size[,free])/last(/vfs.fs.size[,total])
```

O Zabbix obterá os valores mais recentes para os espaços livre e total do disco e calculará a porcentagem de acordo com a fórmula dada.

Exemplo 2

Cálculo da média de 10 minutos do número de valores processados pelo Zabbix.

Uso da função **avg**:

```
avg(/Zabbix Server/zabbix[wcache,values],10m)
```

Note que o uso extensivo de itens calculados com longos períodos de tempo pode afetar a performance do Zabbix Server.

Exemplo 3

Cálculo de banda de rede total em eth0.

Soma de duas funções:

```
last(/net.if.in[eth0,bytes])+last(/net.if.out[eth0,bytes])
```

Exemplo 4

Cálculo da porcentagem de tráfego de entrada.

Expressão mais complexa:

```
100*last(/net.if.in[eth0,bytes])/(last(/net.if.in[eth0,bytes])+last(/net.if.out[eth0,bytes]))
```

Veja também: **Exemplos de cálculos agregados**

Cálculos agregados

Visão geral

Em **cálculos** agregados a informação de vários itens pode ser coletada pelo Zabbix Server (pela realização de consultas diretas ao banco de dados) para calcular uma agregação, dependendo da função agregada utilizada.

Cálculos agregados não requerem qualquer agente em execução no host sendo monitorado.

Para definir cálculos agregados, selecione o tipo de item **Calculado**.

Sintaxe

Agregados podem ser recuperados trabalhando com:

- histórico de itens:

```
aggregate_function(function(/host/item,parameter),function(/host2/item2,parameter),...)
```

- uma **função foreach** como único parâmetro:

`aggregate_function(foreach_function(/*/key?[group="host group"],timeperiod))`

onde:

- `aggregate_function` é uma das **funções agregadas** suportadas: `avg`, `max`, `min`, `sum`, etc.
- `foreach_function` é uma das funções `foreach` suportadas: `avg_foreach`, `count_foreach`, etc.

Funções `foreach` usam um filtro de item, para operar com o histórico de múltiplos itens, e retornam um array de valores - um para cada item.

Note:

Se o agregado resultar em um valor numérico flutuante (float) ele será aparado para um inteiro se o tipo de informação do item é Numérico (unsigned).

Um cálculo agregado pode se tornar não suportado se:

- nenhum dos itens referenciados foi encontrado (o que pode ocorrer se a chave do item está incorreta, nenhum dos itens existe ou todos os grupos incluídos estão incorretos)
- nenhum dado para calcular uma função

Exemplos de uso

Exemplos de chaves para cálculos agregados.

Exemplo 1

Espaço em disco total do grupo de host 'MySQL Servers'.

```
sum(last_foreach(/*/vfs.fs.size[/,total]?[group="MySQL Servers"]))
```

Exemplo 2

Soma dos valores mais recentes de todos os itens correspondentes a `net.if.in[*]` no host.

```
sum(last_foreach(/host/net.if.in[*]))
```

Exemplo 3

Média de carga de processador (load) do grupo de host 'MySQL Servers'.

```
avg(last_foreach(/*/system.cpu.load[,avg1]?[group="MySQL Servers"]))
```

Exemplo 4

Média de 5 minutos do número de consultas por segundo para grupo de host 'MySQL Servers'.

```
avg(avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m))
```

Exemplo 5

Média de carga de CPU (load) de todos os hosts em múltiplos grupos de host que possuem etiquetas específicas.

```
avg(last_foreach(/*/system.cpu.load?[(group="Servers A" or group="Servers B" or group="Servers C") and (tag="servers")]))
```

Exemplo 6

Cálculo usado nas somas dos valores mais recentes de um grupo de host inteiro.

```
sum(last_foreach(/*/net.if.out[eth0,bytes]?[group="video"])) / sum(last_foreach(/*/nginx_stat.sh[active]?[group="video"]))
```

Exemplo 7

O número total de itens não suportados no grupo de host 'Zabbix servers'.

```
sum(last_foreach(/*/zabbix[host,,items_unsupported]?[group="Zabbix servers"]))
```

Examples of correct/incorrect syntax

Expressions (including function calls) cannot be used as history, trend, or `foreach` **function** parameters. However, those functions themselves can be used in other (non-historical) function parameters.

Expression	Example
Valid	<pre>avg(last(/host/key1),last(/host/key2)*10,last(/host/key1)*100) max(avg(avg_foreach(/*/system.cpu.load?[group="Servers A"],5m)),avg(avg_foreach(/*/system.cpu.load?[group="Servers B"],5m)),avg(avg_foreach(/*/system.cpu.load?[group="Servers C"],5m)))</pre>

Expression	Example
Invalid	<code>sum(/host/key,10+2)</code> <code>sum(/host/key, avg(10,2))</code> <code>sum(/host/key,last(/host/key2))</code>

Note that in an expression like:

```
sum(sum_foreach(/resptime[*],5m))/sum(count_foreach(/resptime[*],5m))
```

it cannot be guaranteed that both parts of the equation will always have the same set of values. While one part of the expression is evaluated, a new value for the requested period may arrive and then the other part of the expression will have a different set of values.

8 Verificações internas

Visão geral

Verificações internas permitem monitorar os processos internos do Zabbix. Em outras palavras, você pode monitorar o que está ocorrendo com o Zabbix Server ou Zabbix Proxy.

Verificações internas são calculadas:

- no Zabbix Server - se o host é monitorado pelo Server
- no Zabbix Proxy - se o host é monitorado pelo Proxy

Verificações internas são processadas pelo Server ou Proxy, independentemente do estado de manutenção do host.

Para usar este item, escolha o tipo de item **Zabbix interno**.

Note:

Verificações internas são processadas pelos poolers do Zabbix.

Desempenho

Usar alguns itens internos pode afetar negativamente o desempenho. Esses itens são:

- `zabbix[host,,items]`
- `zabbix[host,,items_unsupported]`
- `zabbix[hosts]`
- `zabbix[items]`
- `zabbix[items_unsupported]`
- `zabbix[queue]`
- `zabbix[requiredperformance]`
- `zabbix[stats,,queue]`
- `zabbix[triggers]`

As seções frontend do usuário **System information** e fila **Queue** também são afetadas.

Verificações suportadas

- Parâmetros sem colchetes angulares são constantes - por exemplo, 'host' e 'available' em `zabbix[host,<type>,available]`. Use-os na key do item como estão.
- Valores para itens e parâmetros de itens que não são suportados em proxy só podem ser obtidos se o host for monitorado pelo servidor. E vice-versa, valores "não suportados no servidor" só podem ser obtidos se o host for monitorado pelo proxy.

Key			
▲	Descrição	Retornar valor	Comentários
<code>zabbix[boottime]</code>	Tempo de inicialização do processo do servidor Zabbix ou do proxy Zabbix em segundos.	Inteiro.	
<code>zabbix[cluster,discovery,nodes]</code>			

Key		
	Descubra os nodes high availability cluster .	JSON.
		Este item pode ser usado em descoberta de baixo nível.
zabbix[history]	Número de valores armazenados na tabela HISTORY.	Inteiro.
		Este item está desativado desde o Zabbix 6.0. Não use se o MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado! (não suportado no proxy)
zabbix[history_log]		

Key		
Número de valores armazenados na tabela HISTORY_LOG.	Inteiro.	Este item está desativado desde o Zabbix 6.0. Não use se o MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado! (não suportado no proxy)
zabbix[history_str]		

Key		
Número de valores armazenados na tabela HISTORY_STR table.	Inteiro.	Este item está desativado desde o Zabbix 6.0. Não use se o MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado!! (não suportado no proxy)
zabbix[history_text]		

Key			
	Número de valores armazenados na tabela HISTORY_TEXT.	Inteiro.	Este item está desativado desde o Zabbix 6.0 MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado!! (não suportado no proxy)
zabbix[history_uint]			

Key		
Número de valores armazenados na tabela HISTORY_UINT.	Inteiro.	TEste item está desativado desde o Zabbix 6.0. Não use se o MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado!! Este item é suportado desde o Zabbix 1.8.3. (não suportado no proxy)
zabbix[host,,items] Número de itens habilitados (suportados e não suportados) no host.	Inteiro.	Este item é suportado desde o Zabbix 3.0.0.
zabbix[host,,items_unsupported]		

Key			
	Número de itens não suportados habilitados no host.	Inteiro.	Este item é suportado desde Zabbix 3.0.0.
zabbix[host,,maintenance]			

Key

Status de manutenção atual de um host.

0 -Status de manutenção atual de um host.,
1 - host em manutenção com coleta de dados,
2 - host em manutenção sem coleta de dados.

Este item é sempre processado pelo servidor do Zabbix independente do local do host (não suportado no proxy). O proxy não receberá esse item com dado de configuração. O segundo parâmetro deve estar vazio e ser reservado para uso futuro.

zabbix[host,discovery,interfaces]

Key	JSON object.	
Detalhes de todas as interfaces configuradas do host no frontend do Zabbix..	JSON object.	Este item pode ser usado em descoberta de baixo nível low-level discovery . Este item é suportado desde o Zabbix 3.4.0. (não suportado no proxy)
zabbix[host,<type>,available]		

Key		
Disponibilidade da interface principal de um tipo específico de verificação no host.	0 - não disponível, 1 - disponível, 2 - desconhecido.	Os tipos válidos são: agent, snmp, ipmi, jmx O valor do item é calculado com base nos parâmetros em relação ao host un-reach-ability/unavailability . Este item é suportado desde o Zabbix 2.0.0.
zabbix[hosts] Número de hosts monitorados.	Inteiro.	
zabbix[items] Número de itens habilitados (suportados e não suportados).	Inteiro.	
zabbix[items_unsupported] Número de itens não suportados.	Inteiro.	
zabbix[java,<param>]		

Key

Informações sobre Zabbix
Java gateway.

Se <param> for ping, retorna "1". Pode ser usado para verificar a disponibilidade do gateway Java usando a função trigger nodata().

Se <param> for version, retorna a versão do Java gateway. Exemplo: "2.0.0".

Os
val-
ores
vál-
idos

param
são:
ping,
ver-
sion

O se-
gundo
parâmetro
deve
estar
vazio
e
reser-
vado
para
uso
fu-
turo.

zabbix[lld_queue]

Key		
Contagem de valores enfileirados na fila de processamento de descoberta de baixo nível.	Inteiro.	Este item pode ser usado para monitorar o comprimento da fila de processamento de descoberta de baixo nível.
		Esse item é suportado desde o Zabbix 4.2.0.
zabbix[preprocessing_queue]		

Key		
Contagem de valores enfileirados na fila de pré-processamento.	Inteiro.	Este item pode ser usado para monitorar o comprimento da fila de pré-processamento.
		Este item é suportado desde o Zabbix 3.4.0.
zabbix[process,<type>,<mode>,<state>]		

Tempo que um processo específico do Zabbix ou um grupo de processos (identificado por <type> e <mode>) passou em <state> em porcentagem. É calculado somente no último minuto.

Se o <mode> para o número de processo do Zabbix que não está em execução (por exemplo, com 5 pollers rodando o <mode> é especificado como 6), tal item se tornará um estado não suportado.

O mínimo e o máximo se referem à porcentagem de uso para um único processo. Portanto, se em um grupo de 3 pollers as porcentagens de uso por processo foram 2, 18 e 66, o mínimo retornaria 2 e o máximo retornaria 66.

Os processos relatam o que estão fazendo na memória compartilhada e o processo de auto-monitoramento resume esses dados a cada segundo. As mudanças de estado (ocupado/inativo) são registradas quando ocorrem - assim, um processo que se torna ocupado é registrado como tal e não muda ou atualiza o estado até se tornar inativo. Isso garante que até mesmo processos totalmente suspensos sejam corretamente registrados como 100% ocupados.

Atualmente, "ocupado" significa "não dormindo", mas no futuro podem ser introduzidos estados adicionais - esperando por bloqueios, realizando consultas ao banco de dados, etc.

No Linux e na maioria dos outros sistemas, a resolução é de 1/100 de segundo.

Porcentagem do tempo.
Float.

tipos
su-
por-
ta-
dos
server
pro-
cesses:
alert
man-
ager,
alert
syncer,
alerter,
avail-
abil-
ity
man-
ager,
con-
figu-
ra-
tion
syncer,
dis-
cov-
erer,
esca-
la-
tor,
ha
man-
ager,
his-
tory
poller,
his-
tory
syncer,
house-
keeper,
http
poller,
icmp
pinger,
ipmi
man-
ager,
ipmi
poller,
java
poller,
lld
man-
ager,
lld
worker,
odbc
poller,
poller,
pre-
pro-
cess-
ing
man-

Key

zabbix[proxy,<name>,<param>]

Key

Informação sobre Zabbix proxy.

Inteiro.

nome:
nome
do
proxy

Os
val-
ores
vál-
di-
dos
para
são:
lastaccess
- do
úl-
timo
heart-
beat
mes-
sage
rece-
bido
do
proxy
delay
-
quanto
tempo
os
val-
ores
cole-
ta-
dos
es-
tão
como
não
envi-
a-
dos,
cal-
cu-
lado
como
"atraso
do
proxy"
(difer-
ença
en-
tre o
tempo
atual
do
proxy
e o
times-
tamp
do
valor
mais
antigo
não
envi-

Key			
zabbix[proxy_history]			
Número de valores na tabela de histórico do proxy esperando para serem enviados para o servidor.	Inteiro.		(não suportado no servidor)
zabbix[queue,<from>,<to>]			
Número de itens monitorados na fila que estão atrasados pelo menos <from> segundos, mas menos que <to> segundos.	Inteiro.		from - padrão: 6 segundos to - padrão: infinito Time-unit symbols (s,m,h,d,w) são suportados por esses parâmetros.
zabbix[rcache,<cache>,<mode>]			

Key			
	Estatísticas de disponibilidade do cache de valores do Zabbix.	Inteiro (por tamanho); float (por porcentagem).	cache: buffer Os modes válidos são: total - tamanho total do buffer free - tamanho do buffer livre pfree - porcentagem do buffer livre used - tamanho do buffer utilizado pused - porcentagem do buffer utilizado O modo pused é suportado desde o Zabbix 4.0.0.
zabbix[requiredperformance]			

Key	Desempenho necessário do servidor Zabbix ou do proxy Zabbix, em novos valores por segundo esperados.	Float.	Correlaciona aproximadamente com "Desempenho do servidor necessário", novos valores por segundo em Reports → System information.
zabbix[stats,<ip>,<port>]			

Key

Métricas internas do servidor
Zabbix ou do proxy Zabbix
remoto.

Objeto JSON.

ip -
Lista
de
IP/DNS/network
mask
dos
servers/proxies
a
serem
con-
sul-
ta-
dos
re-
mo-
ta-
mente
(o
padrão
e de
127.0.0.1)
port
-
Porta
do
servi-
dor/proxy
a ser
con-
sul-
tada
re-
mo-
ta-
mente
(o
padrão
é
10051)

Perceba
que
a so-
lici-
tação
de
sta-
tus
so-
mente
será
aceita
pe-
los
en-
dereços
lista-
dos
em
'Stat-
sAI-
lowedIP'
server/proxy
pa-
ram-

Key

zabbix[stats,<ip>,<port>,queue,<from>,<to>]

Key

Métricas de fila interna do
servidor Zabbix ou do proxy
Zabbix remoto (consulte
zabbix[queue,<from>,<to>]).

JSON object.

ip -
Lista
de
IP/DNS/network
mask
de
servers/proxies
a ser
con-
sul-
tada
re-
mo-
ta-
mente
(o
padrão
é
127.0.0.1)
port
-
Porta
do
servi-
dor/proxy
a ser
con-
sul-
tada
re-
mo-
ta-
mente
(o
padrão
é
10051)
de -
de-
layed
by at
least
(o
padrão
é de
6 se-
gun-
dos)
para
- de-
layed
by at
most
(de-
fault
is
infin-
ity)

Perceba
que
a so-
lici-
tação
de

Key

zabbix[tcache,cache,<parameter>]

Key

Estatísticas de eficácia do
cache de tendências do
Zabbix..

Inteiro (por tamanho; float
(por porcentagem)).

Valid
**pa-
ram-
e-
ters**
are:
all -
número
total
de
solic-
i-
tações
cache
(padrão)
hits -
cache
hits
phits
- per-
cent-
agem
dos
cache
hits
misses
-
cache
per-
di-
dos
pmisses
- por-
cent-
agem
dos
cache
per-
di-
dos
items
- n'
requests
-
número
de
solic-
i-
tações
cached
pitems
- per-
cent-
age
of
cached
items
from
cached
items
+ re-
quests.
Por-
cent-
agem

Key			
zabbix[trends]			
	Número de valores armazenados na tabela TRENDS.	Inteiro.	Este item está ultrapassado desde o Zabbix 6.0. Não suporta MySQL InnoDB, Oracle ou PostgreSQL se tiver sendo usado! (não suportado no proxy)
zabbix[trends_uint]			

Key		
Número de valores armazenados na tabela TRENDS_UINT.	Inteiro.	Este item está depreciado desde o Zabbix 6.0. Não use se MySQL InnoDB, Oracle ou PostgreSQL estiver sendo usado! Este item é suportado desde o Zabbix 1.8.3. (não suportado no proxy)
zabbix[triggers] Número de triggers habilitados no banco de dados do Zabbix, com todos os itens habilitados nos hosts habilitados.	Integer.	(não suportado no proxy)
zabbix[uptime] Tempo de atividade do processo do servidor Zabbix ou do proxy Zabbix em segundos.	Inteiro.	
zabbix[vcache,buffer,<mode>]		

Key			
	Estatísticas de disponibilidade do cache de valores do Zabbix.	Inteiro (por tamanho); float (por porcentagem).	Os modes válidos são: total - tamanho total do buffer free - tamanho do buffer livre pfree - porcentagem do buffer livre used - tamanho do buffer utilizado pused - porcentagem do buffer utilizado (não suportado no proxy)
	zabbix[vcache,cache,<parameter>]		

Key

Estatísticas de eficácia do
cache de valores do Zabbix.

Inteiro.

Com o parâmetro mode:
0 - normal mode,
1 - mode memoria baixa

Os
val-
ores
vál-
idos
do
parâmetro
são:
solicitações
-
número
total
de
solic-
i-
tações
hits -
número
de
cache
hits
(his-
tory
val-
ues
taken
from
the
cache)
misses
-
número
de
caches
ausentes
(his-
tory
val-
ues
taken
from
the
database)
mode
-
value
cache
oper-
ating
mode

Esse
item
é su-
por-
tado
desde
o
Zab-
bix
2.2.0
o
parâmetro
mode
desde

Key			
zabbix[version]	Versão do servidor ou proxy Zabbix.	String.	Este item é suportado desde o Zabbix 5.0.0.
			Exemplo de valor re-tornável: 5.0.0beta1
zabbix[vmware,buffer,<mode>]	Estatísticas de disponibilidade do cache do Zabbix para VMware..	Inteiro (por tamanho); float (por porcentagem).	Os modes válidos são: <ul style="list-style-type: none"> total - tamanho total do buffer free - tamanho do buffer livre pfree - porcentagem do buffer livre used - tamanho do buffer utilizado pused - porcentagem do buffer utilizado
zabbix[wcache,<cache>,<mode>]			

Estatísticas e disponibilidade
do cache de escrita do
Zabbix.

É
obri-
gatório
es-
peci-
ficar
o
<cache>.

Cache
values

Mode
all
(padrão)

Número total de valores
processados pelo servidor ou
proxy Zabbix, exceto para
itens não suportados.

Inteiro

Counter.
Você
pode
usar
essa
chave
com
a
etapa
de
pré-
processamento
Change
per
sec-
ond
para
obter
es-
tatís-
ticas
de
val-
ores
por
se-
gundo.
Counter.

float

Número de valores float
processados.

Inteiro

uint

Número de valores processed
unsigned integer values.

Inteiro

Counter.

str

Número de valores de
caracteres/string
processados.

Inteiro

Counter.

log

Número de valores de log
processados.

Inteiro

Counter.

text

Número de valores de textos
processados.

Inteiro

Counter.

not
supported

Número de vezes que o
processamento do item
resultou em item não
suportado ou permaneceu
em seu estado.

Inteiro

Counter.

Key

history

pfree
(padrão)Porcentagem do buffer de
histórico livre.

Float.

O cache de histórico é utilizado para armazenar valores de itens. Um número baixo indica problemas de performance no banco de dados.

free

Tamanho do buffer de
histórico livre.

Inteiro

total

Tamanho total do buffer de
histórico.

Integer

used

Tamanho do buffer de
histórico utilizado.

Inteiro

pused

Porcetagem do buffer de
histórico utilizado.

Float.

O modo pused é suportado desde do Zabbix 4.0.0.

Key				
index	pfree (padrão)	Tamanho do Historico de índice de buffer.	Float.	History in- dex cache is used to in- dex val- ues stored in his- tory cache. O cache In- dex é su- por- tado desde o Zab- bix 3.0.0.
	free	Tamanho do histórico do índice de buffer de histórico livre.	Inteiro	
	total	Tamanho total do histórico do índice de buffer de histórico.	Inteiro	
	used	Tamanho do histórico do índice de buffer de histórico utilizado.	Inteiro	
	pused	Porcentagem do histórico do índice de buffer de histórico utilizado.	Float.	O modo pused é su- por- tado desde o Zab- bix 4.0.0.

Key				
trend	pfree (padrão)	Porcentagem do trend cache livre.	Float.	A memória Trend cache armazenada para a hora atual para todos os itens que recebem dados. (não suportado no proxy (não suportado no proxy))
	free	Tamanho trend buffer livre.	Inteiro	
	total	Tamanho total do trend buffer.	Inteiro	
	used	Tamanho do trend buffer utilizado.	Inteiro	

Key				
	pu	Porcentagem do trend buffer utilizado.	Float.	(não suportado no proxy)
	used			O modo pu
				used é su
				por-
				tado
				desde
				o
				Zab-
				bix
				4.0.0.

9 Verificações SSH

Visão geral

Verificações SSH são executadas como monitoramento sem agente. O Zabbix Agent não é necessário para verificações SSH.

Para realizar verificações SSH o Zabbix Server deve ser inicialmente **configurado** com suporte a SSH2 (libssh2 ou libssh). Veja também: [Requisitos](#).

Attention:

Apenas libssh é suportado a partir do RHEL/CentOS 8.

Configuração

Autenticação por senha

As verificações SSH oferecem dois métodos de autenticação, um com par de usuário/senha e outro baseado em arquivo de chave.

Se você não pretende usar chaves, nenhuma configuração adicional é necessária, além de associar a libssh2/libssh ao Zabbix, se você estiver compilando a partir dos fontes.

Autenticação por arquivo de chave

Para usar autenticação baseada em chave para itens SSH, certas alterações nas configurações do Server são necessárias.

Abra o arquivo de configuração do Zabbix Server (**zabbix_server.conf**) como root e procure pela seguinte linha:

```
# SSHKeyLocation=
```

Descomente-a e informe o caminho completo para a pasta onde as chaves pública e privada estão localizadas (ou estarão, caso ainda não criadas):

```
SSHKeyLocation=/home/zabbix/.ssh
```

Salve o arquivo e reinicie o zabbix_server em seguida.

No comando acima, /home/zabbix é o diretório pessoal do usuário zabbix e .ssh é um diretório (oculto) onde por padrão as chaves pública e privada serão geradas por um comando **ssh-keygen** executado dentro do diretório pessoal.

Comumente os pacotes de instalação do zabbix-server de diferentes distribuições de SO criam a conta de usuário zabbix com um diretório pessoal em lugares não muito comuns e conhecidos (diferente das contas do sistema), p.e. /var/lib/zabbix.

Antes de iniciar a criação das chaves, uma abordagem para realocar o diretório pessoal para um local melhor conhecido (mais intuitivo) pode ser considerada. Isto corresponderá com a configuração do parâmetro SSHKeyLocation do Zabbix Server mencionada acima.

Estes passos podem ser pulados se a conta zabbix foi adicionada manualmente de acordo com a **seção de instalação** porque neste caso muito provavelmente o diretório já estará localizado em /home/zabbix.

Para alterar a configuração para a conta de usuário zabbix todos os processos em execução que a estão usando devem ser parados:

```
# service zabbix-agent stop
# service zabbix-server stop
```

Para mudar a localização do diretório pessoal com uma tentativa de movê-lo (caso ele exista) o seguinte comando deve ser executado:

```
# usermod -m -d /home/zabbix zabbix
```

É absolutamente possível que um diretório pessoal não exista no antigo local (no CentOS por exemplo), então ele deve ser criado no novo local. Uma tentativa segura de fazê-lo é:

```
# test -d /home/zabbix || mkdir /home/zabbix
```

Para certificar-se de que tudo está seguro, comandos adicionais podem ser executados para configurar as permissões do novo diretório pessoal:

```
# chown zabbix:zabbix /home/zabbix
# chmod 700 /home/zabbix
```

Os processos previamente parados agora podem ser reiniciados:

```
# service zabbix-agent start
# service zabbix-server start
```

Agora as etapas de criação das chaves pública e privada podem ser executadas pelo comando:

```
# sudo -u zabbix ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/zabbix/.ssh/id_rsa):
Created directory '/home/zabbix/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/zabbix/.ssh/id_rsa.
Your public key has been saved in /home/zabbix/.ssh/id_rsa.pub.
The key fingerprint is:
90:af:e4:c7:e3:f0:2e:5a:8d:ab:48:a2:0c:92:30:b9 zabbix@it0
The key's randomart image is:
+---[ RSA 2048]-----+
|          |
|          .  |
|         o   |
|        .o   |
|+         .S  |
|.+.    o =   |
|E .    * =   |
|=o .  ..* .  |
|... oo.o+   |
+-----+

```

Nota: as chaves pública e privada (id_rsa.pub e id_rsa respectivamente) foram geradas (acima) por padrão no diretório /home/zabbix/.ssh que corresponde à configuração do parâmetro SSHKeyLocation do Zabbix Server.

Attention:

Tipos de chave diferentes de "rsa" podem ser suportados pelo utilitário ssh-keygen e servidores SSH, mas elas podem não ser suportadas pela libssh2, usada pelo Zabbix.

Configuração do shell

Esta etapa deve ser executada apenas uma vez para cada host que será monitorado por verificações SSH.

Usando o comando abaixo, o arquivo de chave **pública** pode ser instalado em um host remoto 10.10.10.10 de modo que então as verificações SSH podem ser executadas com a conta de usuário root:

```
# sudo -u zabbix ssh-copy-id root@10.10.10.10
A authenticity of host '10.10.10.10 (10.10.10.10)' não pode ser estabelecida.
RSA key fingerprint é 38:ba:f2:a4:b5:d9:8f:52:00:09:f7:1f:75:cc:0b:46.
Você tem certeza que quer continuar com a conexão (sim/não)? sim
Atenção: Adicionado permanentemente '10.10.10.10' (RSA) à lista de hosts conhecidos.
```

senha para root@10.10.10.10':
Agora tente fazer login na máquina, com "ssh 'root@10.10.10.10'", e verifique:
 .ssh/authorized_keys
para garantir que não adicionamos chaves extras que você não estava esperando.
Agora é possível verificar o login SSH usando a chave privada padrão (/home/zabbix/.ssh/id_rsa) para a conta de usuário zabbix:
sudo -u zabbix ssh root@10.10.10.10
Se o login for bem-sucedido, então a parte de configuração do shell está finalizada e a conexão remota SSH pode ser encerrada.

Configuração de item

Os comandos reais a serem executados devem ser informados no campo **Script executado** na configuração do item.
Múltiplos comandos podem ser executados um após o outro colocando-os em uma nova linha. Neste caso os valores retornados também estarão formatados como multilinha.

ItemTagsPreprocessing

* NameSSH test check (without passphrase)

TypeSSH agent

* Keyssh.run[clear]Select

Type of informationText

Host interface10.10.10.10:10050

Authentication methodPublic key

* User nameroot

* Public key fileid_rsa.pub

* Private key fileid_rsa

Key passphrase

* Executed scriptservice mysql-server status

* Update interval1m

Todos os campos obrigatórios estão marcados com um asterisco vermelho.
Os campos que requerem informação específica para os itens SSH são:

Parâmetro	Descrição	Comentários
Tipo	Selecione Agente SSH aqui.	
Chave	Chave de item única (por host) no formato ssh.run[<descrição curta única>,<ip>,<porta>,<codificação>]	<descrição curta única> é obrigatória e deve ser única para todos os itens SSH por host A porta padrão é 22, não a porta especificada na interface à qual este item está associado
Método de autenticação	Um de "Senha" ou "Chave pública"	
Nome de usuário	Nome de usuário para autenticação no host remoto. Obrigatório	

Parâmetro	Descrição	Comentários
Arquivo de chave pública	Nome do arquivo de chave pública se Método de autenticação é "Chave pública". Obrigatório	Exemplo: id_rsa.pub - nome de arquivo de chave pública padrão gerado pelo comando ssh-keygen
Arquivo de chave privada	Nome do arquivo de chave privada se Método de autenticação é "Chave pública". Obrigatório	Exemplo: id_rsa - nome de arquivo de chave privada padrão
Senha ou Chave de senha (passphrase)	Senha para autenticar ou Chave de senha se tiver sido usada para a chave privada	Deixe o campo Chave de senha vazio se não tiver sido usada uma Consulte também problemas conhecidos em relação ao uso de chave de senha
Script executado	Comando(s) shell executado usando a sessão remota SSH	Exemplos: date +%s service mysql-server status ps auxww grep httpd wc -l

Attention:

A biblioteca libssh2 pode truncar scripts executáveis para ~32kB.

10 Verificações Telnet

Visão Geral

As verificações de Telnet são realizadas como monitoramento sem agente. O agente Zabbix não é necessário para as verificações de Telnet.

Configurable fields

O(s) comando(s) real(is) a ser(em) executado(s) deve(m) ser colocados no campo **Script executado** na configuração do item. Múltiplos comandos podem ser executados um após o outro colocando-os em uma nova linha. Neste caso, o valor retornado também será formatado como multilinhas.

Caracteres suportados com os quais o prompt do shell pode terminar:

- \$
- #
- >
- %

Note:

Uma linha de prompt do telnet que termina com um desses caracteres será removida do valor retornado, mas apenas para o primeiro comando na lista de comandos, ou seja, apenas no início da sessão telnet.

Chave	Descrição
telnet.run[<unique short description>,<ip>,<port>,<encoding>]	Executa um comando em um dispositivo remoto usando conexão telnet.

Attention:

Se uma verificação telnet retornar um valor com caracteres não-ASCII e em uma codificação não-UTF8, então o parâmetro <encoding> da chave deve ser especificado corretamente. Para mais detalhes, consulte a seguinte página [encoding of returned values](#).

11 Verificações externas

Visão geral

Verificação externa é uma verificação executada pelo Zabbix Server pela **execução de um script shell** ou um binário. No entanto, quando hosts são monitorados por um Zabbix Proxy, as verificações externas são executadas pelo Proxy.

As verificações externas não requerem qualquer agente em execução no host sendo monitorado.

A sintaxe da chave do item é:

```
script[<parameter1>,<parameter2>,...]
```

Onde:

ARGUMENTO	DEFINIÇÃO
script	Nome de um script shell ou um binário.
parâmetro(s)	Parâmetros de linha de comando opcionais.

Se você não quer passar nenhum parâmetro para o script você pode usar:

```
script[] ou  
script
```

O Zabbix Server consultará o diretório definido como localização para scripts externos (parâmetro 'ExternalScripts' no **arquivo de configuração do Zabbix Server**) e executará o comando. O comando será executado com o mesmo usuário com o qual o Zabbix Server está sendo executado, então quaisquer permissões de acesso ou variáveis de ambiente devem ser manipuladas em scripts agrupados (wrapper script), se necessário, e as permissões para o comando devem permitir execução para o usuário. Apenas comandos no diretório especificado ficam disponíveis para execução.

Warning:

Não abuse de verificações externas! Como cada script requer iniciar um processo filho (fork) pelo Zabbix Server, a execução de muitos scripts pode diminuir muito a performance do Zabbix.

Exemplo de uso

Executando o script **check_oracle.sh** com o primeiro parâmetro '-h'. O segundo parâmetro será substituído pelo endereço IP ou nome DNS, dependendo da seleção nas propriedades do host.

```
check_oracle.sh["-h","{HOST.CONN}"]
```

Assumindo que o host está configurado para usar o endereço IP, o Zabbix irá executar:

```
check_oracle.sh '-h' '192.168.1.4'
```

Resultado de verificação externa

O valor de retorno da verificação externa é a saída padrão junto com a saída de erro padrão (a saída completa com espaços em branco finais removidos é retornada desde o Zabbix 2.0).

Attention:

Um item texto (tipo de informação caracter, log ou texto) não se tornará não suportado no caso de haver dados na saída de erro padrão.

No caso de o script requisitado não ser encontrado ou o Zabbix Server não possuir permissões para executá-lo, o item se tornará não suportado e a mensagem de erro correspondente será apresentada. No caso de tempo esgotado, o item será marcado também como não suportado, uma mensagem de erro relacionada será apresentada e o processo filho para o script será destruído.

12 Itens de trapper

Visão geral

Itens de captura (trapper) aceitam entrada de dados em vez de consultá-los.

É útil para qualquer dado que você queira "colocar" para dentro do Zabbix.

Para usar um item de captura (trapper) você deve:

- ter um item de captura (trapper) configurado no Zabbix
- enviar o dado para dentro do Zabbix

Configuração

Item configuration

To configure a trapper item:

- Go to: Configuration → Hosts
- Click on Items in the row of the host
- Click on Create item
- Enter parameters of the item in the form

The screenshot shows the 'Item' configuration form in Zabbix. The 'Item' tab is selected. The form contains the following fields:

- * Name:** Trapper item
- Type:** Zabbix trapper (dropdown menu)
- * Key:** trap
- Type of information:** Text (dropdown menu)
- * History storage period:** Do not keep history (radio button), Storage period (radio button, selected), 3600 (input field)

All mandatory input fields are marked with a red asterisk.

The fields that require specific information for trapper items are:

Configuração do item

Para configurar um item de captura (trapper):

- Acesse Configuração → Hosts
- Clique em Itens na linha do host
- Clique em Criar item
- Insira os parâmetros do item no formulário

This is a duplicate of the screenshot above, showing the Zabbix item configuration form for a trapper item with the same fields and values.

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Os campos que requerem informações específicas para itens de trapper são:

Tipo	Selecione o Zabbix trapper aqui.
Chave	Insira uma chave que será utilizada para reconhecer o item ao enviar dados.
Tipo de informação	Selecione o tipo de informação que corresponderá ao formato dos dados que serão enviados.

Hosts permitidos	<p>Lista de endereços IP delimitados por vírgula, opcionalmente em notação CIDR, ou nomes DNS. Se especificado, as conexões de entrada serão aceitas apenas dos hosts listados aqui. Se o suporte IPv6 estiver ativado, então '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' são tratados igualmente e '::/0' permitirá qualquer endereço IPv4 ou IPv6. '0.0.0.0/0' pode ser usado para permitir qualquer endereço IPv4. Observe que os "endereços "IPv4-compatible IPv6" (0000::/96 prefix) são suportados, mas são depreciados pelo RFC4291. Por exemplo: 127.0.0.1, 192.168.1.0/24, 192.168.3.1-255, 192.168.1-10.1-255, ::1, 2001:db8::/32, mysqlserver1, zabbix.example.com, {HOST.HOST} Espaços e user macros são permitidos neste campo desde o Zabbix 2.2.0. Macros do host {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN} são permitidos neste campo desde o Zabbix 4.0.2.</p>
------------------	--

Note:

Você pode ter que esperar até 60 segundos após salvar o item até que o servidor pegue as alterações de uma atualização do cache de configuração, antes de poder enviar valores.

Enviando dados

No caso mais simples, podemos usar o utilitário **zabbix_sender** para enviar um 'valor de teste':

```
zabbix_sender -z <server IP address> -p 10051 -s "New host" -k trap -o "test value"
```

Para enviar o valor, usamos estas chaves:

- z - para especificar o endereço IP do servidor Zabbix
- p - para especificar o número da porta do servidor Zabbix (10051 - padrão)
- s - para especificar o host (certifique-se de usar o nome 'técnico' **host name** aqui, em vez do nome 'visível')
- k - para especificar a chave do item que acabamos de definir
- o - para especificar o valor real a ser enviado

Attention:

O processo de trapper do Zabbix não expande macros usadas na chave do item na tentativa de verificar a existência da chave do item correspondente para o host de destino.

Visualização

Este é o resultado em Monitoramento → Últimos dados:

≡ Latest data

<div> <div><</div> <div>🔍</div> </div> <div>Subfilter affects only filtered data</div>				
<div>HOSTS</div> <div>New host 1</div>				
<div>DATA</div> <div>With data Without data</div>				
<input type="checkbox"/> Host	Name ▲	Last check	Last value	Change
<input type="checkbox"/> New host	Trapper item	2m 27s	test value	

Observe que se um único valor numérico for enviado, o gráfico de dados mostrará uma linha horizontal à esquerda e à direita do ponto de tempo do valor.

13 Monitoramento JMX

Visão geral

Monitoramento JMX pode ser usado para monitorar contadores JMX de uma aplicação Java.

O monitoramento JMX possui suporte nativo no Zabbix na forma de um daemon (serviço) chamado "Zabbix Java Gateway", introduzido desde o Zabbix 2.0.

Para resgatar o valor de um contador JMX em particular em um host, o Zabbix Server consulta o Zabbix **Java Gateway**, que por sua vez usa a [API de gerenciamento JMX](#) para consultar a aplicação de interesse remotamente.

Para mais detalhes e configuração verifique a seção [Zabbix Java Gateway](#).

Warning:

A comunicação entre o Java Gateway e a aplicação JMX monitorada deve estar devidamente liberada pelo firewall de rede.

Habilitando monitoramento JMX remoto para aplicação Java

Uma aplicação Java não necessita de qualquer software adicional instalado, mas precisa ser iniciada com as opções de linha de comando especificadas abaixo para ter suporte ao monitoramento JMX remoto.

No mínimo, se você deseja apenas iniciar o monitoramento de uma aplicação Java simples em um host local sem aplicação de segurança, inicie com estas opções:

```
java \
-Dcom.sun.management.jmxremote \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=false \
-Dcom.sun.management.jmxremote.ssl=false \
-Dcom.sun.management.jmxremote.registry.ssl=false \
-jar /usr/share/doc/openjdk-6-jre-headless/demo/jfc/Notepad/Notepad.jar
```

Isto faz com que o Java ouça conexões JMX de entrada na porta 12345, apenas do host local, e declara que não é necessário autenticação ou SSL.

Se você deseja permitir conexões em outra interface, configure o parâmetro `-Djava.rmi.server.hostname` com o IP dessa interface.

Se você deseja ser mais rigoroso com a segurança, há muitas outras opções de Java disponíveis. Como exemplo, a instrução abaixo inicia a aplicação com um conjunto de opções mais versátil e a deixa aberta para uma rede mais ampla, não apenas o host local.

```
java \
-Djava.rmi.server.hostname=192.168.3.14 \
-Dcom.sun.management.jmxremote \
-Dcom.sun.management.jmxremote.port=12345 \
-Dcom.sun.management.jmxremote.authenticate=true \
-Dcom.sun.management.jmxremote.password.file=/etc/java-6-openjdk/management/jmxremote.password \
-Dcom.sun.management.jmxremote.access.file=/etc/java-6-openjdk/management/jmxremote.access \
-Dcom.sun.management.jmxremote.ssl=true \
-Dcom.sun.management.jmxremote.registry.ssl=true \
-Djavax.net.ssl.keyStore=$YOUR_KEY_STORE \
-Djavax.net.ssl.keyStorePassword=$YOUR_KEY_STORE_PASSWORD \
-Djavax.net.ssl.trustStore=$YOUR_TRUST_STORE \
-Djavax.net.ssl.trustStorePassword=$YOUR_TRUST_STORE_PASSWORD \
-Dcom.sun.management.jmxremote.ssl.need.client.auth=true \
-jar /usr/share/doc/openjdk-6-jre-headless/demo/jfc/Notepad/Notepad.jar
```

A maioria destas (se não todas) configurações pode ser especificada em `/etc/java-6-openjdk/management/management.properties` (ou onde quer que esse arquivo esteja em seus sistema).

Note que se você deseja usar SSL, você tem que modificar o script `startup.sh` com a adição das opções `-Djavax.net.ssl.*` para o Java Gateway, de forma que ele saiba onde encontrar o 'keystore' e o 'truststore'.

Consulte [Monitoramento e Gerenciamento Usando JMX](#) para uma descrição detalhada.

Configurando interfaces e itens JMX no Zabbix Frontend

Com o Java Gateway em execução, o Server tendo conhecimento de onde o encontrar e uma aplicação Java iniciada com suporte a monitoramento JMX remoto, é hora de configurar as interfaces e itens no GUI do Zabbix.

Configurando interface JMX

Você começa criando uma interface do tipo JMX no host de interesse.

Host Templates IPMI Tags Macros Inventory Encryption Value mapping

* Host name JMX host

Visible name JMX host

* Groups Java (new) X type here to search Select

Interfaces	Type	IP address	DNS name	Connect to	Port
Agent		127.0.0.1		IP DNS	10050
JMX		127.0.0.1		IP DNS	12345

[Add](#)

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Adicionando um item de agente JMX

Para cada contador JMX no qual você está interessado você adiciona um item **Agente JMX** anexado a essa interface.

A chave na imagem abaixo diz `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`.

Item Tags Preprocessing

* Name Used heap memory

Type JMX agent

* Key `jmx["java.lang:type=Memory","HeapMemoryUsage.used"]`

Type of information Numeric (unsigned)

* Host interface 127.0.0.1:12345

* JMX endpoint `service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi`

User name {JMX_USERNAME}

Password {JMX_PASSWORD}

Units

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Os campos que requerem informação específica para itens JMX são:

Tipo
Chave

Configure **Agente JMX** aqui.

A chave de item `jmx[]` contém três parâmetros:

- nome de objeto** - o nome de objeto de um MBean
- nome de atributo** - um nome de atributo MBean com campos de dados compostos opcionais separados por pontos
- descrição curta única** - uma descrição única que permita múltiplos itens JMX com o mesmo nome de objeto e nome de atributo no host (opcional)

Veja abaixo mais detalhes em relação às chaves de item JMX. Desde o Zabbix 3.4, você pode descobrir MBeans e atributos de MBean usando o item de [descoberta de baixo-nível](#) `jmx.discovery[]`.

Endpoint JMX	Você pode especificar um endpoint JMX customizado. Certifique-se de que os parâmetros de conexão ao endpoint JMX correspondem à interface JMX. Isto pode ser realizado com o uso das macros {HOST.*} como feito no endpoint JMX padrão.
Nome de usuário	Este campo é suportado desde o 3.4.0. Macros {HOST.*} e macros de usuário são suportadas. Especifique o nome de usuário, se você tiver configurado autenticação em sua aplicação Java.
Senha	Macros de usuário são suportadas. Informe a senha, se você tiver configurado autenticação em sua aplicação Java. Macros de usuário são suportadas.

Se você deseja monitorar um contador booleano que é ou "verdadeiro" ou "falso", então você deve informar o tipo de informação como "Numérico (unsigned)" e utilizar a etapa de pré-processamento "Booleano para decimal" na aba de pré-processamento. O Zabbix Server armazenará os valores Booleanos como 1 ou 0, respectivamente.

Chaves de item JMX em mais detalhes

Atributos simples

Um nome de objeto MBean é nada senão uma string que você define em sua aplicação Java. Um nome de atributo, por outro lado, pode ser mais complexo. No caso de o atributo retornar um tipo de dado primitivo (um inteiro, uma string, etc.) não há nada com o que se preocupar, a chave se parecerá com esta:

```
jmx[com.example:Type=Hello,weight]
```

Neste exemplo temos um nome de objeto "com.example:Type=Hello", o nome de atributo é "weight" e provavelmente o tipo de valor retornado deve ser "Numérico (float)".

Atributos retornando dados compostos

Se torna mais complicado quando seu atributo retorna dado composto. Por exemplo: seu nome de atributo é "laranja" e ele retorna um hash representando seus parâmetros, como "peso", "cor", etc. Sua chave pode se parecer como esta:

```
jmx[com.example:Type=Hello,laranja.peso]
```

Assim é como um nome de atributo e uma chave hash são separados, usando um símbolo de ponto. Da mesma forma, se um atributo retorna dados compostos aninhados as partes são separadas por um ponto:

```
jmx[com.example:Type=Hello,frutas.laranja.peso]
```

Atributos retornando dado tabular

Atributos de dado tabular consistem de um ou múltiplos atributos compostos. Se tal atributo é especificado no parâmetro de nome de atributo então este valor de item retornará a estrutura completa do atributo em formato JSON. Os valores de elemento individuais dentro do atributo de dado tabular podem ser recuperados usando pré-processamento.

Exemplo de atributo de dado tabular:

```
jmx[com.example:type=Hello,comidainfo]
```

Valor do item:

```
[
  {
    "a": "laranja",
    "b": "banana",
    "c": "cereja"
  },
  {
    "a": "batata",
    "b": "alface",
    "c": "cebola"
  }
]
```

Problemas com pontos

Até aqui tudo bem. Mas e se um nome de atributo ou uma chave hash contiver um símbolo de ponto? Aqui temos um exemplo:

```
jmx[com.example:Type=Hello,todas.frutas.laranja.peso]
```

Isto é um problema. Como informar ao Zabbix que o nome de atributo é "todas.frutas", não apenas "todas"? Como distinguir um ponto que é parte do nome de um ponto que separa um nome de atributo e chaves hash?

Antes da **2.0.4** o Zabbix Java Gateway era incapaz de lidar com tais situações e os usuários ficavam com itens **NÃO SUPORTADOS**. Desde a versão 2.0.4 isto é possível, e tudo o que você precisa fazer é escapar os pontos que fazem parte do nome com uma contrabarra:

```
jmx[com.example:Type=Hello,todas\.frutas.laranja.peso]
```

Da mesma forma, se sua chave hash contém um ponto escape-o:

```
jmx[com.example:Type=Hello,todas\.frutas.laranja.peso\.total]
```

Outros problemas

Um caracter de contrabarra em um nome de atributo deve ser escapado:

```
jmx[com.example:type=Hello,c:\\documentos]
```

Para lidar com quaisquer outros caracteres especiais em uma chave de item JMX, por favor consulte a [seção](#) de formato de chave de item.

Isto é na verdade tudo o que temos. Feliz monitoramento JMX!

Tipos de dados não-primitivos

Desde o Zabbix 4.0.0 é possível trabalhar com MBeans personalizados retornando tipos de dados não-primitivos, o que sobrescreve o método **toString()**.

Usando endpoint customizado com JBoss EAP 6.4

Endpoints customizados permitem trabalhar com diferentes protocolos de transporte que não o RMI padrão.

Para ilustrar esta possibilidade, vamos tentar configurar o monitoramento de um JBoss EAP 6.4 como exemplo. Primeiro, vamos supor algumas coisas:

- Você já tem o Zabbix Java Gateway instalado. Se não, então você pode fazê-lo de acordo com a [documentação](#).
- O Zabbix Server e Java Gateway estão instalados com o prefixo `/usr/local/`
- O JBoss já está instalado em `/opt/jboss-eap-6.4/` e em execução no modo standalone
- Devemos assumir que todos estes componentes estão operando no mesmo host
- Firewall e SELinux estão desabilitados (ou configurados de acordo)

Vamos fazer algumas configurações simples no `zabbix_server.conf`:

```
JavaGateway=127.0.0.1
StartJavaPollers=5
```

E no arquivo de configuração `zabbix_java/settings.sh` (ou `zabbix_java_gateway.conf`):

```
START_POLLERS=5
```

Garanta que o JBoss ouve sua porta de gerenciamento padrão:

```
$ netstat -natp | grep 9999
tcp        0      0 127.0.0.1:9999          0.0.0.0:*               LISTEN      10148/java
```

Agora vamos criar um host com interface JMX 127.0.0.1:9999 no Zabbix.

Host Templates IPMI Tags Macros Inventory Encryption Value mapping

* Host name

Visible name

* Groups

Java (new) ×

type here to search

Select

Interfaces	Type	IP address	DNS name	Connect to	Port
Agent		<input type="text" value="127.0.0.1"/>	<input type="text"/>	<div>IP DNS</div>	<input type="text" value="10050"/>
JMX		<input type="text" value="127.0.0.1"/>	<input type="text"/>	<div>IP DNS</div>	<input type="text" value="9999"/>

[Add](#)

Como sabemos que esta versão do JBoss usa o protocolo JBoss Remoting em vez do RMI, nós podemos atualizar em massa o parâmetro de endpoint JMX para os itens em nosso modelo JMX de acordo:

```
service:jmx:remoting-jmx://{HOST.CONN}:{HOST.PORT}
```

Mass update

Item Tags Preprocessing

Type ☐ Original

JMX endpoint ☒

Vamos atualizar o cache de configuração:

```
$ /usr/local/sbin/zabbix_server -R config_cache_reload
```

Note que você pode encontrar um erro primeiro.

```
3. mc [root@centos7-dev]:/home/vagrant/zabbix-3.2.6/src/zabbix_java (ssh)
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
    at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
    at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
    at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
    at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
    at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
    at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
    at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
    ... 3 common frames omitted
2017-11-07 13:52:12.644 [pool-1-thread-1] WARN com.zabbix.gateway.SocketProcessor - error processing request
com.zabbix.gateway.ZabbixException: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
    at com.zabbix.gateway.JMXItemChecker.getValues(JMXItemChecker.java:97) ~[zabbix-java-gateway-3.4.2.jar:na]
    at com.zabbix.gateway.SocketProcessor.run(SocketProcessor.java:63) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1149) [na:1.8.0_144]
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624) [na:1.8.0_144]
    at java.lang.Thread.run(Thread.java:748) [na:1.8.0_144]
Caused by: java.net.MalformedURLException: Unsupported protocol: remoting-jmx
    at javax.management.remote.JMXConnectorFactory.newJMXConnector(JMXConnectorFactory.java:359) ~[na:1.8.0_144]
    at javax.management.remote.JMXConnectorFactory.connect(JMXConnectorFactory.java:269) ~[na:1.8.0_144]
    at com.zabbix.gateway.ZabbixJMXConnectorFactory$1.run(ZabbixJMXConnectorFactory.java:76) ~[zabbix-java-gateway-3.4.2.jar:na]
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511) ~[na:1.8.0_144]
    at java.util.concurrent.FutureTask.run(FutureTask.java:266) ~[na:1.8.0_144]
    ... 3 common frames omitted
2017-11-07 13:52:14.889 [Thread-0] INFO com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885) as stopped
2017-11-07 13:52:26.167 [main] INFO com.zabbix.gateway.JavaGateway - Zabbix Java Gateway 3.4.2 (revision 72885) has started
```

"Protocolo não suportado: remoting-jmx (Unsupported protocol: remoting-jmx)" significa que o Java Gateway não sabe como trabalhar com o protocolo especificado. Isso pode ser resolvido com a criação de um arquivo ~/modulos_necessarios.txt com o seguinte conteúdo:

```
jboss-as-remoting
jboss-logging
jboss-logmanager
jboss-marshalling
jboss-remoting
jboss-sasl
jcl-over-slf4j
jul-to-slf4j-stub
log4j-jboss-logmanager
remoting-jmx
slf4j-api
xnio-api
xnio-nio</pre>
```

e então executando o comando:

```
$ for i in $(cat ~/modulos_necessarios.txt); do find /opt/jboss-eap-6.4 -iname ${i}*.jar -exec cp {} /usr/
```

Assim, o Java Gateway terá todos os módulos necessários para trabalhar com jmx-remoting. O que nos resta é reiniciar o Java Gateway, aguardar uns instantes e se você fez tudo certo, ver os dados de monitoramento JMX começando a chegar no Zabbix (veja também: [Último dados](#)).

14 Monitoramento ODBC

Visão geral

O monitoramento ODBC corresponde ao tipo de item Monitor de banco de dados no Zabbix Frontend.

ODBC é uma API (middleware de linguagem de programação C) para acesso a Sistemas de Gerenciamento de Banco de Dados (SGBD). O conceito ODBC foi desenvolvido pela Microsoft e mais tarde portado para outras plataformas.

O Zabbix pode consultar qualquer banco de dados que seja suportado pelo ODBC. Para isso, o Zabbix não conecta diretamente o banco de dados, mas usa a interface ODBC e drivers configurados no ODBC. Esta função permite monitoramento mais eficiente de diferentes bancos de dados para múltiplos propósitos - por exemplo, verificar filas de banco de dados específicas, estatísticas de uso e assim por diante. O Zabbix suporta unixODBC, que é uma das implementações de API ODBC de código aberto mais comumente usadas.

Attention:

Consulte também os [problemas conhecidos](#) para verificações ODBC.

Instalando o unixODBC

A forma sugerida de instalação do unixODBC é usar o repositório de pacotes padrão do sistema Linux. Nas distribuições Linux mais populares o unixODBC é incluído no repositório de pacotes por padrão. Se não estiver disponível, ele pode ser obtido na página principal do unixODBC: <http://www.unixodbc.org/download.html>.

Para instalar o unixODBC nos sistemas baseados em RedHat/Fedora usando o gerenciador de pacotes yum:

```
shell> yum -y install unixODBC unixODBC-devel
```

Para instalar em sistemas baseados em SUSE usando o gerenciador de pacotes zypper:

```
# zypper in unixODBC-devel
```

Note:

O pacote unixODBC-devel é necessário para compilar o Zabbix com suporte a unixODBC.

Instalando drivers unixODBC

Um driver de banco de dados unixODBC deve ser instalado para o banco de dados que será monitorado. O unixODBC tem uma lista de bancos de dados e drivers suportados em: <http://www.unixodbc.org/drivers.html>. Em algumas distribuições Linux os drivers de banco de dados estão inclusos nos repositórios de pacotes.

Instalando driver de banco de dados MySQL nos sistemas baseados em RedHat/Fedora usando o gerenciador de pacotes yum:

```
shell> yum install mysql-connector-odbc
```

Instalando driver de banco de dados MySQL nos sistemas baseados em SUSE usando o gerenciador de pacotes zypper:

```
zypper in MyODBC-unixODBC
```


Configurando unixODBC

A configuração do ODBC é feita pela edição dos arquivos **odbcinst.ini** e **odbc.ini**. Para verificar a localização do arquivo, digite:

```
shell> odbcinst -j
```

O **odbcinst.ini** é usado para listar os drivers de banco de dados ODBC instalados:

```
[mysql]
Description = ODBC for MySQL
Driver      = /usr/lib/libmyodbc5.so
```

Detalhes dos parâmetros:

Atributo	Descrição
mysql	Nome do driver de banco de dados.
Description	Descrição do driver de banco de dados.
Driver	Localização da biblioteca do driver de banco de dados.

O **odbc.ini** é usado para definir as fontes de dados (bancos de dados que serão consultados):

```
[test]
Description = MySQL test database
Driver      = mysql
Server      = 127.0.0.1
User        = root
Password    =
Port        = 3306
Database    = zabbix
```

Detalhes dos parâmetros:

Atributo	Descrição
test	Nome da fonte de dados (DSN).
Description	Descrição da fonte de dados.
Driver	Nome do driver de banco de dados - como definido em odbcinst.ini
Server	Endereço IP/nome DNS do servidor de banco de dados.
User	Usuário para conexão no banco de dados.
Password	Senha para o usuário acima.
Port	Porta de conexão ao banco de dados.
Database	Nome do banco de dados.

Para verificar se a conexão ODBC está funcionando corretamente, uma conexão ao banco de dados deve ser testada. Isto pode ser feito com o utilitário **isql** (incluso no pacote unixODBC):

```
shell> isql test
+-----+
| Connected!                                |
|                                           |
| sql-statement                            |
| help [tablename]                         |
| quit                                     |
|                                           |
+-----+
SQL>
```

Compilando o Zabbix com suporte a ODBC

Para habilitar o suporte a ODBC, o Zabbix deve ser compilado com a seguinte opção:

```
--with-unixodbc[=ARG] use driver odbc em conjunto com pacote unixODBC
```

Note:

Veja mais sobre instalação do Zabbix a partir do [código fonte](#).

Configuração de item no Zabbix Frontend

Configure um **item** de monitoramento de banco de dados.

Item

Tags

Preprocessing

*

Name

MySQL host count

Type

Database monitor

▼

*

Key

db.odbc.select[mysql-simple-check,test]

Type of information

Numeric (unsigned)

▼

User name

zabbix

Password

*

SQL query

select count(*) from hosts

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Especificamente para itens de monitoramento de banco de dados você deve informar:

Tipo	Selecione Monitor de banco de dados aqui.
Chave	<p>Informe uma de duas chaves de item suportadas:</p> <p>db.odbc.select[<descrição curta única>,<dsn>,<string de conexão>] - este item é desenhado para retornar apenas um valor, p.e. a primeira coluna da primeira linha do resultado de uma consulta SQL. Se uma consulta retornar mais do que uma coluna, apenas a primeira coluna é lida. Se uma consulta retornar mais de uma linha, apenas a primeira linha é lida.</p> <p>db.odbc.get[<descrição curta única>,<dsn>,<string de conexão>] - este item é capaz de retornar múltiplas linhas/colunas em formato JSON. Assim ele pode ser usado como um item principal que coleta todos os dados em uma chamada de sistema, enquanto o pré-processamento JSONPath pode ser usado em itens dependentes para extrair valores individuais. Para mais informações, veja um exemplo de formato retornado, usado em descoberta de baixo-nível. Este item é suportado desde o Zabbix 4.4.</p> <p>A descrição única servirá para identificar o item em gatilhos, etc.</p> <p>Apesar de dsn e string de conexão serem parâmetros opcionais, ao menos um deles deve estar presente. Se ambos (nome da fonte de dados - DSN e string de conexão) forem definidos, o DSN será ignorado.</p> <p>O nome da fonte de dados, se usado, deve ser configurado como especificado em odbc.ini.</p> <p>A string de conexão pode conter argumentos específicos de driver.</p> <p>Exemplo (conexão para driver MySQL ODBC 5):</p> <p>=> db.odbc.get[MySQL exam- ple,, "Driver=/usr/local/lib/libmyodbc5a.so;Database=master;Server=127.0.0.1"]</p>

Nome de usuário	Informe o nome do usuário do banco de dados Este parâmetro é opcional se um usuário tiver sido especificado no <code>odbc.ini</code> . Se é usada string de conexão, e Nome de usuário não estiver vazio, ele é adicionado à string de conexão como <code>UID=<user></code>
Senha	Informe a senha para o usuário acima Este parâmetro é opcional se a senha estiver definida em <code>odbc.ini</code> . Se é usada string de conexão, e o campo Senha não estiver vazio, ela é adicionada à string de conexão como <code>PWD=<password></code> .
Consulta SQL	Informe a consulta SQL (query). Note que com o item <code>db.odbc.select []</code> a consulta deve retornar apenas um valor.
Tipo de informação	É importante saber que tipo de informação será retornado pela consulta, de modo que seja selecionado corretamente aqui. Com um tipo de informação incorreto o item se tornará não suportado.

Notas importantes

- Itens de monitoramento de banco de dados se tornarão são suportados se nenhum processo `odbc poller` for iniciado na configuração do Server ou Proxy. Para ativar os pollers ODBC, defina o parâmetro `StartODBCPollers` no arquivo de configuração do Zabbix **Server** ou, para verificações pelo Proxy, no arquivo de configuração do **Proxy**.
- O Zabbix não limita o tempo de execução de consultas (query). Cabe ao usuário escolher consultas que sejam executadas em um período de tempo razoável.
- O valor de parâmetro de **timeout** do Zabbix Server é usado como tempo de expiração de sessão ODBC - ODBC login timeout (note que dependendo dos drivers ODBC o tempo de expiração de login deve ser ignorado).
- O comando SQL deve retornar um conjunto de resultado como qualquer consulta com `select ...`. A sintaxe de consulta dependerá do SGBD que a processará. A sintaxe de requisição para uma 'stored procedure' deve ser iniciada com a palavra-chave `call`.

Mensagens de erro

Mensagens de erro ODBC são estruturadas em campos para prover informação detalhada. Por exemplo:

Cannot execute ODBC query: [SQL_ERROR]: [42601] [7] [ERROR: syntax error at or near ";"; Error while executing

	Código de erro nativo	Mensagem de erro nativa
	SQLState	
Mensagem Zabbix	Código de retorno ODBC	

Note que o comprimento da mensagem de erro é limitado a 2048 bytes, então a mensagem pode ser truncada. Se houver mais de um registro de diagnóstico ODBC o Zabbix tenta concatená-los (separados com `|`) tanto quanto o limite de comprimento permitir.

1 Configurações UnixODBC recomendadas para MySQL

Instalação

*** Red Hat Enterprise Linux/CentOS**:

```
# yum install mysql-connector-odbc
```

***Debian/Ubuntu**:

Por favor tome como referência a [documentação do MySQL](#) para baixar o driver de banco de dados necessário para a plataforma correspondente.

Para algumas informações adicionais consulte a [instalação do unixODBC](#).

Configuração

A configuração do ODBC é feita pela edição dos arquivos **odbcinst.ini** e **odbc.ini**. Estes de arquivos de configuração podem ser encontrados em `/etc`. O arquivo **odbcinst.ini** pode estar ausente e neste caso é necessário criá-lo manualmente.

odbcinst.ini

```
[mysql]
Description = General ODBC for MySQL
Driver       = /usr/lib64/libmyodbc5.so
Setup       = /usr/lib64/libodbcmyS.so
FileUsage   = 1
```

Por favor considere os seguintes exemplos de configuração dos parâmetros do **odbc.ini**.

- Um exemplo com conexão por IP:

```
[TEST_MYSQL]
Description = Banco de dados MySQL 1
Driver = mysql
Port = 3306
Server = 127.0.0.1
```

- Um exemplo com conexão por IP e uso de credenciais. Um banco de dados Zabbix é usado por padrão:

```
[TEST_MYSQL_FILLED_CRED]
Description = Banco de dados MySQL 2
Driver = mysql
User = root
Port = 3306
Password = zabbix
Database = zabbix
Server = 127.0.0.1
```

- Um exemplo com conexão por socket e uso de credenciais. Um banco de dados Zabbix é usado por padrão:

```
[TEST_MYSQL_FILLED_CRED SOCK]
Description = Banco de dados MySQL 3
Driver = mysql
User = root
Password = zabbix
Socket = /var/run/mysqld/mysqld.sock
Database = zabbix
```

Todas as outras opções de parâmetro de configuração podem ser encontradas na [documentação oficial do MySQL](#).

2 Configurações UnixODBC recomendadas para PostgreSQL

Instalação

- **** Red Hat Enterprise Linux/CentOS**:**

```
# yum install postgresql-odbc
```

- **Debian/Ubuntu:**

Por favor considere como referência a [documentação do PostgreSQL](#) para baixar o driver de banco de dados necessário para a plataforma correspondente.

Para informações adicionais, consulte a [instalação do unixODBC](#).

Configuração

A configuração do ODBC é feita editando os arquivos **odbcinst.ini** e **odbc.ini**. Estes arquivos de configuração podem ser encontrados em /etc. O arquivo **odbcinst.ini** pode estar ausente e neste caso é necessário criá-lo manualmente.

Considere os seguintes exemplos:

odbcinst.ini

```
[postgresql]
Description = ODBC geral para PostgreSQL
Driver       = /usr/lib64/libodbcpsql.so
Setup       = /usr/lib64/libodbcpsqlS.so
FileUsage   = 1
```

Desde a versão 1.6 se o gerenciador de driver foi construído com suporte a thread, você pode adicionar o

```
# Esta entrada altera o nível padrão de serialização de thread.  
Threading = 2
```

odbc.ini

```
[TEST_PSQL]  
Description = Banco de dados PostgreSQL 1  
Driver = postgresql  
#CommLog = /tmp/sql.log  
Username = zbx_test  
Password = zabbix  
# Nome do servidor. IP ou DNS  
Servername = 127.0.0.1  
# Nome do banco de dados  
Database = zabbix  
# Porta de escuta do Postmaster  
Port = 5432  
# Banco de dados é somente leitura  
# Se a fonte de dados permitirá atualizações.  
ReadOnly = No  
# Protocolo de backend do PostgreSQL  
# Note que quando usando conexões SSL esta configuração é ignorada.  
# 7.4+: Use o protocolo 7.4(V3). Isto é compatível apenas com backends 7.4 e maiores.  
Protocol = 7.4+  
# Inclui o OID em SQLColumns  
ShowOidColumn = No  
# Simula um índice único no OID  
FakeOidIndex = No  
# Versionamento de linha  
# Permite que aplicações detectem se os dados podem ser modificados por outros usuários  
# enquanto você está tentando atualizar uma linha.  
# Também acelera o processo de atualização desde que cada coluna única não necessita ser  
# especificada na cláusula WHERE para atualizar uma linha.  
RowVersioning = No  
# Mostra SystemTables  
# O driver tratará as tabelas de sistema como tabelas regulares em SQLTables. Isto é com  
# para Access de forma que você pode ver as tabelas de sistema.  
ShowSystemTables = No  
# Se verdadeiro, o driver automaticamente usa declaração de cursor/fetch para manipular os comandos SELECT  
Fetch = Yes  
# Booleans como Char  
# Booleans são mapeados para SQL_CHAR, caso contrário para SQL_BIT.  
BooleansAsChar = Yes  
# Modo SSL  
SSLmode = Yes  
# Envia para backend na conexão  
ConnSettings =  
  
3 Configurações UnixODBC recomendadas para Oracle
```

Instalação

Por favor, consulte [Oracle documentation](#) para todas instruções necessárias.

Para mais informações, consulte: [Installing unixODBC](#).

4 Configurações UnixODBC recomendadas para MSSQL

Instalação

- **** Red Hat Enterprise Linux/CentOS**:**

```
# yum -y install freetds unixODBC
```

- **Debian/Ubuntu:**

Por favor tome como referência o [guia de usuário FreeTDS](#) para baixar o driver de banco de dados necessário para a plataforma correspondente.

Para algumas informações adicionais consulte a [instalação do unixODBC](#).

Configuração

A configuração do ODBC é feita pela edição dos arquivos **odbcinst.ini** e **odbc.ini**. Estes de arquivos de configuração podem ser encontrados em /etc. O arquivo **odbcinst.ini** pode estar ausente e neste caso é necessário criá-lo manualmente.

Por favor considere os seguintes exemplos:

odbcinst.ini

```
$ vi /etc/odbcinst.ini
[FreeTDS]
Driver = /usr/lib64/libtdsodbc.so.0
```

odbc.ini

```
$ vi /etc/odbc.ini
[sql1]
Driver = FreeTDS
Server = <IP do servior SQL 1>
PORT = 1433
TDS_Version = 8.0
```

15 Itens dependentes

Visão geral

Há situações quando um item junta múltiplas métricas de uma vez ou até faz mais sentido coletar métricas relacionadas de forma simultânea, por exemplo:

- Utilização de CPU de cores individuais
- Tráfego de entrada/saída/total de rede

Para permitir a coleta de métricas em massa e uso simultâneo em vários itens relacionados, o Zabbix suporta itens dependentes. Itens Dependentes dependem de um item principal que coleta seus dados simultaneamente, em uma consulta. Um novo valor para um item principal automaticamente popula os valores dos itens dependentes. Itens Dependentes não têm um intervalo de atualização diferente do item principal.

As opções de pré-processamento do Zabbix podem ser usadas para extrair a parte que é necessária para o item dependente a partir dos dados do item principal.

O pré-processamento é gerenciado por um processo gerenciador de pré-processamento, que foi adicionado no Zabbix 3.4, junto com outros operadores que executam as etapas de pré-processamento. Todos os valores (com ou sem pré-processamento) de diferentes coletores de dados passam através do gerenciador de pré-processamento antes de serem adicionados ao cache de histórico. Comunicação IPC baseada em socket (Socket-based) é usada entre os coletores de dados (pollers, trappers, etc.) e os processos de pré-processamento.

O Zabbix Server ou Zabbix Proxy (se o host é monitorado por Proxy) estão executando etapas de pré-processamento e processando itens dependentes.

Um item de qualquer tipo, até mesmo um item dependente, pode ser configurado como um item principal. Níveis adicionais de itens dependentes podem ser usados para extrair partes menores de valores de um item dependente existente.

Limitações

- Apenas dependência no mesmo host (template) são permitidas
- Um protótipo de item pode depender de outro protótipo de item ou de um item regular do mesmo host
- A contagem máxima de itens dependentes para um item principal é limitada a 29999 (independente do número de níveis de dependência)
- O máximo de 3 níveis de dependência são permitidos
- Um item dependente em um host com um item principal de template não será exportado para XML

Configuração de item

Um item dependente depende de seu item principal para os dados. Este é o porquê o **item principal** deve ser configurado (ou existir) primeiro:

- Vá até: Configuração → Hosts
- Clique em Itens na linha do host
- Clique em Criar item
- Informe os parâmetros do item no formulário

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Clique em Adicionar para salvar o item principal.

Então você pode configurar um **item dependente**.

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Os campos que requerem informação específica para itens dependentes são:

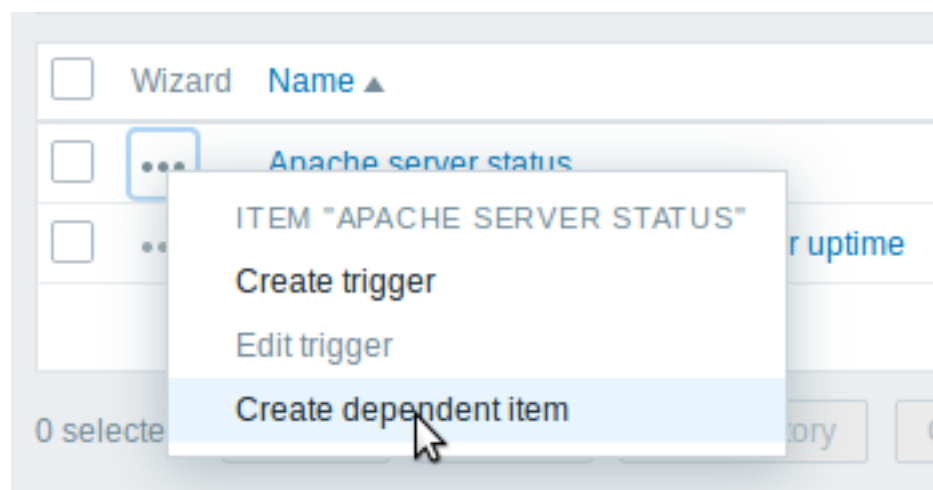
Tipo	Selecione Item dependente aqui.
Chave	Informe a chave que será usada para reconhecer o item.
Item principal	Selecione o item principal. O valor do item principal será usado para popular o valor do item dependente.
Tipo de informação	Selecione o tipo de informação que corresponderá ao formato do dado que será armazenado.

Você pode usar o **pré-processamento** de valor do item para extrair a parte necessária do valor do item principal.

Sem pré-processamento, o valor do item dependente será exatamente o mesmo que o valor do item principal.

Clique em Adicionar para salvar o item dependente.

Um atalho para a criação mais rápida de um item dependente é usar o assistente na lista de item:



Apresentação

Na lista de item os itens dependentes são apresentados com o nome do seu item principal como prefixo.

<input type="checkbox"/>	Name ▲	Triggers	Key
<input type="checkbox"/>	...	Apache server status	web.page.get[127.0.0.1/server-status]
<input type="checkbox"/>	...	Apache server status: Apache server uptime	apache.server.uptime

Se um item principal é excluído, então todos os itens dependentes também o são.

16 Agente HTTP

Visão geral

Este tipo de item permite sondagem de dados usando o protocolo HTTP/HTTPS. Retenção (trapping) também é possível usando o Zabbix Sender ou o protocolo Zabbix Sender.

Uma verificação de item HTTP é executada pelo Zabbix Server. No entanto, quando os host são monitorados por um Zabbix Proxy, as verificações de item HTTP são executadas pelo Proxy.

As verificações de item HTTP não requerem nenhum agente em execução no host sendo monitorado.

O agente HTTP suporta ambos HTTP e HTTPS. O Zabbix seguirá redirecionamentos opcionalmente (veja a opção Seguir redirecionamento abaixo). O número máximo de redirecionamentos é fixo no código em 10 (usando a opção `CURL CURLOPT_MAXREDIRS`).

Veja também [problemas conhecidos](#) na utilização do protocolo HTTPS.

Attention:

O Zabbix Server/Proxy deve ser inicialmente configurado com suporte a cURL (libcurl).

Configuração

Para configurar um item HTTP:

- Vá até: Configuração → Hosts
- Clique em Itens na linha do host
- Clique em Criar item
- Informe os parâmetros do item no formulário

Item Tags Preprocessing

* Name

HTTP agent item

Type

HTTP agent

* Key

http_value_search

Type of information

Numeric (unsigned)

* URL

http://localhost:5200/in/values/_search

Query fields

Name	Value
scroll	10s

Add

Request type

POST

* Timeout

3s

Request body type

Raw data JSON data XML data

Request body

```
{
  "query": {
    "bool": {
      "must": {
        "term": {
          "item_id": "28275"
        }
      }
    }
  }
}
```

Headers

Name	Value
name	value

Add

Required status codes

200

Follow redirects

☒

Retrieve mode

Body Headers Body and headers

Convert to JSON

☐

HTTP proxy

localhost:8080 password@example.com:port

HTTP authentication

None

SSL verify peer

☐

SSL verify host

☐

SSL certificate file

SSL key file

SSL key password

* Host interface

127.0.0.1:9550

Units

* Update interval

1m

Custom intervals

Type	Interval
Enable	Scheduling

Add

* History storage period

Do not keep history Storage period 90d

* Trend storage period

Do not keep trends Storage period 365d

Value mapping

Type here to search

Enable trapping

☐

Populates host inventory field

None

Description

Enabled

☒

Add Test Cancel

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Os campos que requerem informação específica para itens HTTP são:

Tipo

Chave

URL

Selecione **Agente HTTP** aqui.

Informe uma chave de item única.

URL na qual conectar e recuperar os dados. Por exemplo:

https://www.example.com

http://www.example.com/download

Nomes de domínio podem ser especificados em caracteres Unicode. Eles são automaticamente convertidos com punycode para ASCII quando executando a verificação HTTP. O botão Processar pode ser usado para separar campos de consulta opcionais (como ?name=Admin&password=mypassword) da URL, movendo os atributos e valores em Campos da consulta com codificação URL automática. Limitado a 2048 caracteres.

Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.

Isto configura a opção do cURL [CURLOPT_URL](#).

Variáveis para a URL (veja acima).

Especificado como pares de atributo e valor.

Valores são codificados para URL automaticamente. Valores de macros são resolvidos e então codificados para URL automaticamente.

Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.

Isto configura a opção do cURL [CURLOPT_URL](#).

Campos da consulta

Tipo de requisição	<p>Selecione o método de requisição: GET, POST, PUT or HEAD</p> <p>O Zabbix não gastará mais do que a quantidade de tempo configurada no processamento da URL (1-60 segundos). Na verdade este parâmetro define o tempo máximo para a realização de conexão a URL e tempo máximo para realizar uma consulta HTTP. No entanto, o Zabbix não gastará mais do que 2 x Tempo limite segundos em uma verificação. Sufixos de tempo são suportados, p.e. 30s, 1m.</p> <p>Macros suportadas: macros de usuário, macros de descoberta de baixo-nível.</p>
Tempo limite	<p>Isto configura a opção do cURL CURLOPT_TIMEOUT.</p> <p>Selecione o tipo de corpo da requisição:</p> <p>Dados brutos - corpo da requisição HTTP customizado, macros são substituídas mas nenhuma codificação é executada</p> <p>Dados JSON - corpo da requisição HTTP em formato JSON. Macros podem ser usadas como string, número, verdadeiro e falso; macros usadas como strings devem ser quotadas com aspas duplas. Valores de macros são resolvidos e então escapados automaticamente. Se "Content-Type" não for especificado nos cabeçalhos então assumirá o valor padrão "Content-Type: application/json"</p> <p>Dados XML - corpo da requisição HTTP em formato XML. Macros podem ser usadas como um nó de texto, atributo ou seção CDATA. Valores de macros são resolvidos e então escapados automaticamente em um nó de texto e atributo. Se "Content-Type" não for especificado nos cabeçalhos então assumirá o valor padrão "Content-Type: application/json"</p>
Tipo de corpo da requisição	<p>Note que a seleção de Dados XML requer libxml2.</p> <p>Informe o corpo da requisição.</p> <p>Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.</p>
Corpo da requisição	
Cabeçalhos	<p>Cabeçalhos HTTP personalizados que serão enviados na execução de uma requisição.</p> <p>Especificado como pares de atributo e valor.</p> <p>Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.</p>
Códigos de estado requeridos	<p>Isto configura a opção do cURL CURLOPT_HTTPHEADER.</p> <p>Lista de códigos de estado HTTP esperados. Se o Zabbix recebe um código que não está na lista, o item se tornará não suportado. Se vazio, nenhuma verificação será executada.</p> <p>Por exemplo: 200,201,210-299</p> <p>Macros suportadas na lista: macros de usuário, macros de descoberta de baixo-nível.</p>
Seguir redirecionamentos	<p>Isto usa a opção do cURL CURLINFO_RESPONSE_CODE.</p> <p>Marque esta caixa para seguir redirecionamentos HTTP.</p> <p>Isto configura a opção do cURL CURLOPT_FOLLOWLOCATION.</p>
Modo de busca	<p>Selecione a parte da resposta que deve ser recuperadas:</p> <p>Corpo - apenas o corpo</p> <p>Cabeçalhos - apenas cabeçalhos</p> <p>Corpo e cabeçalhos - corpo e cabeçalhos</p>

Converter para JSON

Cabeçalhos são salvos como pares de atributo e valor sob a chave "cabeçalho".

Se 'Content-Type: application/json' é encontrada então o corpo é salvo como um objeto, caso contrário ele é armazenado como string, por exemplo:

```
{
  "header": {
    "<key>": "<value>",
    "<key2>": "<value>"
  },
  "body": <body>
}
```

Proxy HTTP

Você pode especificar um proxy HTTP para usar, usando o formato

[protocol://][username[:password]@]proxy.example.com[:port]

O prefixo opcional protocol:// pode ser usado para especificar protocolos proxy alternativos (p.e. https, socks4, socks5; veja [documentação](#); o suporte ao prefixo de protocolo foi adicionado no cURL 7.21.7). Sem nenhum protocolo especificado, o proxy será tratado como um proxy HTTP. Se você especificar um protocolo errado, a conexão falhará e o item se tornará não suportado.

Por padrão, a porta 1080 será usada.

Se especificado, o proxy irá sobrescrever as variáveis de ambiente de proxy relacionadas como http_proxy, HTTPS_PROXY. Se não especificado, o proxy não irá sobrescrever as variáveis de ambiente de proxy relacionadas. O valor informado é passado "como é", nenhum teste de é executado.

Note que apenas autenticação simples é suportada com proxy HTTP.

Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.

Isto configura a opção do cURL [CURLOPT_PROXY](#).

Tipo de autenticação:

Nenhum - autenticação não usada.

Básica - autenticação básica é usada.

NTLM - autenticação NTLM ([Windows NT LAN Manager](#)) é usada.

Kerberos - autenticação Kerberos é usada. Veja também: [Configuração do Kerberos com Zabbix](#).

Digest - autenticação Digest é usada.

A seleção de um método de autenticação oferecerá dois campos adicionais para informar um nome de usuário e senha, onde macros de usuário e macros de descoberta de baixo-nível são suportadas.

Isto configura a opção do cURL [CURLOPT_HTTPAUTH](#).

Marque esta caixa para verificar o certificado SSL do servidor web. O certificado do servidor será automaticamente obtido do local da autoridade de certificação do sistema (CA). Você pode sobrescrever o local dos arquivos CA usando o parâmetro de configuração SSLCALocation do Zabbix Server ou Proxy.

Isto configura a opção do cURL [CURLOPT_SSL_VERIFYPEER](#).

Autenticação HTTP

verificar par SSL

Verificar host SSL	<p>Marque esta caixa para verificar que o campo Nome Comum ou Nome Alternativo do Sujeito (SAN) do certificado do servidor web corresponde.</p> <p>Isto configura a opção do cURL CURLOPT_SSL_VERIFYHOST.</p>
Arquivo de certificado SSL	<p>Nome do arquivo de certificado SSL usado para autenticação do cliente. O arquivo de certificado deve estar no formato PEM¹. Se o arquivo de certificado também contém a chave privada, deixe este campo vazio. Se a chave está criptografada, informe a senha no campo de senha da chave SSL. O diretório contendo estes arquivos é especificado pelo parâmetro de configuração SSLCertLocation do Zabbix Server ou Proxy.</p> <p>Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.</p>
Arquivo de chave SSL	<p>Isto configura a opção do cURL CURLOPT_SSLCERT.</p> <p>Nome do arquivo de chave privada SSL usado para autenticação do cliente. O arquivo de chave privada deve estar no formato PEM¹. O diretório contendo estes arquivos é especificado pelo parâmetro de configuração SSLKeyLocation do Zabbix Server ou Proxy.</p> <p>Macros suportadas: {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG}, macros de usuário, macros de descoberta de baixo-nível.</p>
Senha da chave SSL	<p>Isto configura a opção do cURL CURLOPT_SSLKEY.</p> <p>senha do arquivo de chave privada SSL.</p> <p>Macros suportadas: macros de usuário, macros de descoberta de baixo-nível.</p>
Habilitar retenção (trapping)	<p>Isto configura a opção do cURL CURLOPT_KEYPASSWD.</p> <p>Com esta caixa marcada, o item também funcionará como um item de retenção (trapper) e aceitará dados enviados para este item pelo Zabbix Sender ou usando protocolo Zabbix Sender.</p>
Host permitidos	<p>Visível apenas se a caixa Habilitar retenção estiver marcada.</p> <p>Lista de endereços de IP separados por vírgula, opcionalmente em notação CIDR, ou nomes de host.</p> <p>Se especificado, conexões de entrada serão aceitas apenas de hosts listados aqui.</p> <p>Se o suporte a IPv6 estiver habilitado então '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' são tratados da mesma forma e '::/0' permitirá qualquer endereço IPv4 ou IPv6. '0.0.0.0/0' pode ser usado para permitir qualquer endereço IPv4.</p> <p>Note, que "endereços IPv6 compatíveis com IPv4" (prefixo 0000::/96) são suportados mas obsoletos pela RFC4291.</p> <p>Exemplo: Server=127.0.0.1, 192.168.1.0/24, 192.168.3.1-255, 192.168.1-10.1-255, ::1,2001:db8::/32, zabbix.domain</p> <p>Espaços e macros de usuário são permitidos neste campo.</p> <p>Macros de host: {HOST.HOST}, {HOST.NAME}, {HOST.IP}, {HOST.DNS}, {HOST.CONN} são permitidas neste campo.</p>

Note:

Se o campo Proxy HTTP for deixada vazia, outra forma de usar um proxy HTTP é configurar as variáveis de ambiente relacionadas ao proxy.

Para HTTP - configure a variável de ambiente `http_proxy` para o usuário do Zabbix Server. Por exemplo:

`http_proxy=http://proxy_ip:proxy_port.`

Para HTTPS - configure a variável de ambiente `HTTPS_PROXY`. Por exemplo:

`HTTPS_PROXY=http://proxy_ip:proxy_port.` Mais detalhes estão disponíveis executando o comando shell: `# man curl`.

Attention:

[1] O Zabbix suporta arquivos de certificado e chave privada apenas no formato PEM. Caso você tenha seu certificado e dados de chave privada em formato de arquivo PKCS #12 (usualmente com extensão *.p12 ou *.pfx) você deve gerar o arquivo PEM dele usando os seguintes comandos:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
```

```
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes -out ssl-cert.key
```

Exemplos

Exemplo 1

Enviar requisições GET simples para obter dados de serviços como Elasticsearch:

- Crie um item GET com URL: `localhost:9200/?pretty`
- Verifique a resposta:

```
{
  "name" : "YQ2VAY-",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "kH4CYqh5QfqgeTsjh2F9zg",
  "version" : {
    "number" : "6.1.3",
    "build_hash" : "af51318",
    "build_date" : "2018-01-26T18:22:55.523Z",
    "build_snapshot" : false,
    "lucene_version" : "7.1.0",
    "minimum_wire_compatibility_version" : "5.6.0",
    "minimum_index_compatibility_version" : "5.0.0"
  },
  "tagline" : "You know, for search"
}
```

- Agora extraia o número de versão usando uma etapa de pré-processamento JSONPath: `$.version.number`

Exemplo 2

Enviar requisições de POST simples para obter dados de serviços como Elasticsearch:

- Crie um item POST com URL: `http://localhost:9200/str/values/_search?scroll=10s`
- Configure o seguinte corpo POST para obter a carga de processador (média de 1 minuto por core)

```
{
  "query": {
    "bool": {
      "must": [{
        "match": {
          "itemid": 28275
        }
      }],
      "filter": [{
        "range": {
          "clock": {
            "gt": 1517565836,
            "lte": 1517566137
          }
        }
      }]
    }
  }
}
```

```

    }
  }
}

• Recebido:

{
  "_scroll_id": "DnF1ZXJ5VGhlbkZldGNoBQAAAAAAAAAaF1lRMlZBWS1UU1pxTmdEeGVwQjRBTfEAAAAAAAAAJRZZUTJWQVktVFN",
  "took": 18,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": 1,
    "max_score": 1.0,
    "hits": [{
      "_index": "dbl",
      "_type": "values",
      "_id": "dqX9VWEBV6sEKSMYk6sw",
      "_score": 1.0,
      "_source": {
        "itemid": 28275,
        "value": "0.138750",
        "clock": 1517566136,
        "ns": 25388713,
        "ttl": 604800
      }
    }]
  }
}

```

- Agora use uma etapa de pré-processamento JSONPath para obter o valor do item: `$.hits.hits[0]._source.value`

Exemplo 3

Verificando se a API Zabbix está disponível, usando [apiinfo.version](#).

- Configuração do item:

Item Tags Preprocessing

* Name

Type

* Key

Type of information

* URL

Query fields

Name	Value
name	value

[Add](#)

Request type

* Timeout

Request body type

Request body

```
{
  "jsonrpc": "2.0",
  "method": "apiinfo.version",
  "params": [],
  "id": 1
}
```

Headers

Name	Value
Content-Type	application/json-rpc

[Add](#)

Required status codes

Follow redirects ☒

Retrieve mode

Note o uso do método POST com dados JSON, configurando cabeçalhos da requisição e solicitando o retorno de apenas cabeçalhos:

- Pré-processamento do valor do item com expressão regular para obter código HTTP:

Item Tags Preprocessing 1

Preprocessing steps

Name	Parameters
1: Regular expression	HTTPV1.1 ([0-9]+)

[Add](#)

- Verificando o resultado nos Últimos dados:

Host groups
Select

Name

Hosts
Select

Show items without data
☐

Application
Select

Show details
☐

Apply
Reset

<input type="checkbox"/> Host	Name ▲	Last check	Last value	Change
<input type="checkbox"/> Zabbix server	- other - (1 Item)			
<input type="checkbox"/>	Check Zabbix API version	2018-05-16 23:50:34	OK (200)	Graph

Exemplo 4

Obtendo informações de clima pela conexão ao serviço público Openweathermap.

- Configure um item principal para coletar dados massivos em um único JSON:

Item
Tags
Preprocessing

* Name

Type

* Key

Type of information

* URL

Query fields

Name	Value
units	metric
lat	{ \$LAT }
lon	{ \$LON }
APPID	{ \$WEATHER_APIKEY }
lang	{ \$WEATHER_LANG }

Add

Request type

* Timeout

Request body type

Request body

Note o uso de macros nos campos da requisição. Tome como referência a [API do Openweathermap](#) para saber como preenchê-los.

Exemplo de JSON retornado como resposta ao agente HTTP:

```
{
  "body": {
    "coord": {
      "lon": 40.01,
```



```

    "lat": 56.11
  },
  "weather": [{
    "id": 801,
    "main": "Clouds",
    "description": "few clouds",
    "icon": "02n"
  }],
  "base": "stations",
  "main": {
    "temp": 15.14,
    "pressure": 1012.6,
    "humidity": 66,
    "temp_min": 15.14,
    "temp_max": 15.14,
    "sea_level": 1030.91,
    "grnd_level": 1012.6
  },
  "wind": {
    "speed": 1.86,
    "deg": 246.001
  },
  "clouds": {
    "all": 20
  },
  "dt": 1526509427,
  "sys": {
    "message": 0.0035,
    "country": "RU",
    "sunrise": 1526432608,
    "sunset": 1526491828
  },
  "id": 487837,
  "name": "Stavrovo",
  "cod": 200
}
}

```

A próxima tarefa é configurar os itens dependentes para extrair dados do JSON.

- Configure uma amostra de item dependente para umidade:

Item	Tags	Preprocessing
		<p>* Name <input type="text" value="Humidity"/></p> <p>Type <input type="text" value="Dependent item"/></p> <p>* Key <input type="text" value="humidity"/></p> <p>Type of information <input type="text" value="Numeric (float)"/></p> <p>* Master item <input type="text" value="Apache: Get weather"/></p> <p>Units <input type="text"/></p>

Outras métricas como 'Temperatura' são adicionados da mesma forma.

- Amostra de item dependente de pré-processamento de valor com JSONPath:

Item
Tags
Preprocessing 1

Preprocessing steps

1:

JSONPath

▼

\$.body.main.humidity

Add

- Verifique o resultado dos dados meteorológicos nos Últimos dados:

▼ <input type="checkbox"/> Host	Name ▲	Inter...	History	Trends	Type	Last check	Last value
▼ <u>weather</u>	Weather (8 Items)						
<input type="checkbox"/>	Get weather get_weather.http	10m	1d		HTTP agent	2018-05-17 01:23:45	{\"body\":{\"coord\":{\"lon...
<input type="checkbox"/>	Get weather HTTP response code get_weather.http_code		7d	0	Depende...	2018-05-17 01:23:45	OK (200)
<input type="checkbox"/>	Humidity humidity		90d	365d	Depende...	2018-05-17 01:23:45	66 %
<input type="checkbox"/>	Temperature temp		90d	365d	Depende...	2018-05-17 01:23:45	15.14 C
<input type="checkbox"/>	Weather weather		90d		Depende...	2018-05-17 01:23:45	Clouds
<input type="checkbox"/>	Weather condition id weather.condition.id		7d	0	Depende...	2018-05-17 01:23:45	801
<input type="checkbox"/>	Weather description weather.description		90d		Depende...	2018-05-17 01:23:45	few clouds
<input type="checkbox"/>	Wind speed wind.speed		90d	365d	Depende...	2018-05-17 01:23:45	1.86 m/s

Exemplo 5

Conexão com página de estado do Nginx e obtenção de suas métricas em massa.

- Configure o Nginx seguindo o [guia oficial](#).
- Configure um item principal para coleção de dados em massa:

Item
Tags
Preprocessing

* Name

Nginx: Get stub status page

Type

HTTP agent

▼

* Key

nginx.get_stub_status

Type of information

Text

▼

* URL

http://{HOST.CONN}/nginx_status

Query fields

Name

Value

name

⇒

value

Add

Request type

GET

▼

* Timeout

3s

Request body type

Raw data

JSON data

XML data

Amostra de saída de estado do Nginx:

Active connections: 1 Active connections:
server accepts handled requests
52 52 52
Reading: 0 Writing: 1 Waiting: 0

A próxima tarefa é configurar os itens dependentes para extrair os dados.

- Configure uma amostra de item dependente para requisições por segundo:

The screenshot shows the 'Preprocessing' configuration for a new item. The fields are as follows:

- Name:** Client requests per second
- Type:** Dependent item
- Key:** nginx_requests_rps
- Type of information:** Numeric (unsigned)
- Master item:** Nginx by HTTP: Nginx: Get stub status page

- Amostra de item de pré-processamento de valor com expressão regular `server accepts handled requests\s+([0-9]+) ([0-9]+) ([0-9]+)`:

The screenshot shows the 'Preprocessing' configuration with two steps:

Preprocessing steps	Name	Parameters
1:	Regular expression	requests\s+([0-9]+) ([0-9]+) ([0-9]+)
2:	Change per second	

- Verifique o resultado completo do módulo de registro (stub) nos Últimos dados:

Host	Name	Last check	Last value
nginx	Nginx (8 Items)		
<input type="checkbox"/>	Accepted client connections	2018-05-18 17:54:53	568
<input type="checkbox"/>	Active connections	2018-05-18 17:54:53	1
<input type="checkbox"/>	Client requests per second	2018-05-18 17:54:53	0 rps
<input checked="" type="checkbox"/>	Get Nginx stub status	2018-05-18 17:54:53	HTTP/1.1 200 OK Se...
<input type="checkbox"/>	Handled connections per second	2018-05-18 17:54:53	0
<input type="checkbox"/>	Reading	2018-05-18 17:54:53	0
<input type="checkbox"/>	Waiting	2018-05-18 17:54:53	0
<input type="checkbox"/>	Writing	2018-05-18 17:54:53	1

17 Verificações Prometheus

Visão geral

O Zabbix pode consultar métricas expostas no formato de linha do Prometheus.

Dois passos são necessários para iniciar a obtenção de dados do Prometheus:

- um **item principal HTTP** apontando para o endpoint de dados apropriado, p.e. `https://<prometheus host>/metrics`
- itens dependentes usando a opção de pré-processamento Prometheus para consultar dados de métricas reunidos pelo item principal

Estas são duas opções de pré-processamento de dados do Prometheus:

- Padrão Prometheus - usado em itens normais para consultar dados Prometheus

- Prometheus para JSON - usado em itens normais e para descoberta de baixo-nível. Neste caso dados Prometheus consultados são retornados em um formato JSON.

Processamento em massa

Processamento em massa é suportado para itens dependentes. Para habilitar o cache e indexação, o pré-processamento Padrão Prometheus deve ser a **primeira** etapa de pré-processamento. Quando o Padrão Prometheus é a primeira etapa de pré-processamento então o dado Prometheus analisado é colocado em cache e indexado pela primeira condição <etiqueta>==<valor> na etapa de pré-processamento Padrão Prometheus. Este cache é reutilizado quando processando outros itens dependentes neste lote. Para otimização de performance, a primeira etiqueta deve ser aquela com mais valores diferentes.

Se há outro pré-processamento a ser feito antes da primeira etapa, ele deve ser movido ou para o item principal ou para um novo item dependente que seria usado como item principal para os itens dependentes relacionados.

Configuração

Considerando que você tenha o item principal HTTP configurado, você precisa criar um **item dependente** que usa uma etapa de pré-processamento Prometheus:

- Informe os parâmetros gerais do item dependente no formulário de configuração
- Vá à aba de Pré-processamento
- Selecione uma opção de pré-processamento Prometheus (Padrão Prometheus ou Prometheus para JSON)

The screenshot shows the 'Preprocessing 1' configuration tab. It features a table with 'Preprocessing steps' and 'Parameters'. The first step is 'Prometheus pattern' with parameters 'cpu_usage_system{cpu="cp' and a dropdown menu showing 'value', 'label', and 'sum'. Below the table is an 'Add' button and a 'Type of information' dropdown set to 'Numeric (unsigned)'.

Preprocessing steps	Name	Parameters
1:	Prometheus pattern	cpu_usage_system{cpu="cp

Below the table, there is an 'Add' button and a 'Type of information' dropdown set to 'Numeric (unsigned)'. To the right of the parameters, there is a dropdown menu with options: value, label, and sum.

Os seguintes parâmetros são especificados para a opção de pré-processamento Padrão Prometheus:

Parâmetro	Descrição	Exemplos
Padrão	<p>Para definir o padrão de dado requerido você pode usar uma linguagem de consulta similar à linguagem de consulta do Prometheus (veja tabela de comparação), p.e.:</p> <p><nome métrica> - selecione por nome da métrica</p> <p>{__name__=<nome métrica>} - selecione por nome da métrica</p> <p>{__name__=~<regex>} - selecione por nome da métrica correspondente a uma expressão regular</p> <p>{<label name>=<valor etiqueta>,...} - selecione por nome da etiqueta (label)</p> <p>{<label name>=~<regex>,...} - selecione por nome da etiqueta correspondente a uma expressão regular</p> <p>{__name__=~".*"}==<valor> - selecione por valor da métrica</p> <p>Ou uma combinação do que está acima:</p> <p><nome métrica>{<label1 name>=<valor etiqueta1>,<label2 name>=~<regex>,...}==<valor></p> <p>O valor da etiqueta pode ser qualquer sequência de caracteres UTF-8, mas os caracteres de contrabarra, aspas duplas e nova linha tem de ser escapados como \\, \" e \n respectivamente; outros caracteres não devem ser escapados.</p>	<pre>wmi_os_physical_memory_free_bytes cpu_usage_system{cpu="cpu-total"} cpu_usage_system{cpu=~".*"} cpu_usage_system{cpu="cpu-total",host=~".*"} wmi_service_state{name="dhcp"}==1 wmi_os_timezone{timezone=~".*"}==1</pre>
Resultado do processamento	<p>Especifique se será retornado o valor, a etiqueta ou aplique a função apropriada (se o padrão corresponde a várias linhas e o resultado precisa ser agregado):</p> <p>valor - retorna o valor da métrica (erro se múltiplas linhas corresponderem)</p> <p>etiqueta - retorna o valor da etiqueta especificada no campo Etiqueta (erro se múltiplas linhas corresponderem)</p> <p>sum - retorna a soma dos valores</p> <p>min - retorna o valor mínimo</p> <p>max - retorna o valor máximo</p> <p>avg - retorna o valor médio</p> <p>count - retorna a contagem de valores</p> <p>Este campo está disponível apenas na opção Padrão Prometheus.</p>	<p>Veja também exemplos de uso de parâmetros abaixo.</p>
Saída	<p>Defina um nome de etiqueta (opcional). Neste caso o valor correspondente ao nome de etiqueta é retornado.</p> <p>Este campo está disponível apenas para a opção Padrão Prometheus, se 'Etiqueta' estiver selecionado no campo Processamento de resultado.</p>	

Exemplos de utilização dos parâmetros

1. O caso de uso mais comum é para retornar o **valor**. Para retornar o valor de /var/db de:

```
node_disk_usage_bytes{path="/var/cache"} 2.1766144e+09<br>node_disk_usage_bytes{path="/var/db"}
20480<br>node_disk_usage_bytes{path="/var/dpkg"} 8192<br>node_disk_usage_bytes{path="/var/empty"}
4096
```

use os seguintes parâmetros:

- Padrão - node_disk_usage_bytes{path="/var/db"}
- Processamento do resultado - selecione 'valor'

2. Você pode também estar interessado no valor **médio** de todos os parâmetros node_disk_usage_bytes:

- Padrão - node_disk_usage_bytes
- Resultado do processamento - selecione 'avg'

3. Enquanto o Prometheus suporta apenas dados numéricos, é popular usar uma contingência que permite retornar uma descrição textual relevante também. Isto pode ser alcançado com um filtro e especificação de uma etiqueta. Então, para retornar o valor da etiqueta 'color' (cor) de

```
elasticsearch_cluster_health_status{cluster="elasticsearch",color="green"} 1  
elasticsearch_cluster_health_status{cluster="elasticsearch",color="yellow"} 0
```

use os seguintes parâmetros:

- Padrão - elasticsearch_cluster_health_status {cluster="elasticsearch"} == 1
- Resultado do processamento - selecione 'etiqueta'
- Etiqueta - especifique 'color'

O filtro (baseado no valor numérico '1') corresponderá a linha apropriada, enquanto a etiqueta retornará a descrição do estado de integridade (atualmente 'verde' (green); mas potencialmente também 'vermelho' (red) ou 'amarelo' (yellow)).

Prometheus para JSON

Dados do Prometheus podem ser usados para descoberta de baixo-nível. Neste caso são necessários dados em formato JSON e a opção de pré-processamento Prometheus para JSON retornará exatamente isso.

Para mais detalhes, consulte [Descoberta usando dados Prometheus](#).

Comparação linguagem de consulta (query language)

A seguinte tabela lista as diferenças e similaridades entre PromQL e a linguagem de consulta de pré-processamento Prometheus do Zabbix.

PromQL	Zabbix
Seletor de Votor Instantâneo	Pré-processamento Prometheus
Diferenças	
AlvoServidor Prometheus para consulta Retorno instantâneo	Texto plano em formato de exposição Prometheus
Operadores de correspondência de etiqueta	Métrica ou valor de etiqueta (Padrão Prometheus) Array de métricas por valor único em JSON (Prometheus para JSON)
Operadores de correspondência de etiqueta	=, !=, =~, !~

PromQL	
Seleção de Vetor Instantâneo	Zabbix Pré-processamento Prometheus
Expressão Regular usada na etiqueta ou nome da métrica correspondente	PCRE
Operação de comparação	Apenas == (igual) é suportado para filtro de valor
Similaridades	
Selecção por nome de métrica igual string	<nome métrica> ou {__name__="<nome métrica>"}
Selecção por nome de métrica correspondente a expressão regular	{__name__=~"<regex>"}

PromQL	
Se-	
le-	
tor	
de	
Ve-	
tor	
In-	
stan-	Zabbix
tâ-	Pré-processamento
neo	Prometheus
Selecção pelo valor <nome etiqueta>="<nome etiqueta>" ,...}	{<label name>="<nome etiqueta>" ,...}
Selecção pelo valor <nome etiqueta>=~"<regex>" ,...}	{<label name>=~"<regex>" ,...}
Selecção por valor que iguala string <__name__=~".*" } == <valor>	{__name__=~".*" } == <valor>

18 Script items

Visão geral

Os itens do Script podem ser usados para coletar o dado executando o código JavaScript para usuário-definido com habilidade para recuperar o dado pelo HTTP/HTTPS. Além do Script, há uma lista de parâmetros (pares de nomes e valor) e intervalo que podem ser específicos.

O tipo de item pode ser útil na coleta dos cenários de dados que exigem múltiplos passos e lógica complexa. Um exemplo, o item Script pode ser configurado para fazer uma chamada HTTP, e então processar o dado recebido de alguma maneira, e passar o valor transformado para a segunda chamada HTTP.

Os itens do Script são processados pelo Zabbix server ou pelo proxy pollers.

Configuração

No campo Tipo do **item configuration form** selecione Script e então preencha os campos requisitados.

Item
Tags
Preprocessing

* Name

Data collector script

Type

Script

* Key

script.data.collector

Type of information

Text

Parameters

Name	Value	Action
host	{HOST.CONN}	Remove
endpoint	{\$ENDPOINT}	Remove
Add		

* Script

var request = new HttpRequest();...

* Timeout

3s

Todos os campos obrigatórios do input são marcados com um asterisco vermelho.

Os campos que requisitam informação específica para os itens do Script são:

Campo	Descrição
Chave	Digite a chave única que irá ser usada para identificar o item.
Parâmetros	Especifique as variáveis a serem passadas para o script como pares de atributos e valores. Built-in macros {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.IP}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}, {ITEM.KEY.ORIG} e user macros são suportados.
Script	Digite o código JavaScript no bloco que aparece quando clicar no campo do parâmetro (ou no botão view/edit próximo a ele). Este código deve providenciar a lógica retornando o valor métrico. O código tem acesso a todos os parâmetros, ele pode performar requisitos HTTP GET, POST, PUT e DELETE tem todos cabeçalhos HTTP e corpo da solicitação. Ver também: Additional JavaScript objects , JavaScript Guide .
Tempo limite	Excedendo o limite de execução do JavaScript(1-60s, padrão 3s); irá retornar erro. Sufixos de tempo são suportados, e.g. 30s, 1m. Dependendo do script pode demorar mais para o tempo limite do trigger.

Exemplos

Coleta de dados simples

Coletar o dado de https://www.example.com/release_notes:

- Criar um item com o tipo "Script".
- No campo do Script, digitar:

```
var request = new HttpRequest();
return request.get("https://www.example.com/release_notes");
```

Coleta de dados com parâmetros

Coletar o conteúdo de uma página específica e fazer uso de parâmetros:

- Criar um item com tipo "Script" e dois parâmetros:
 - **url** : **{\${DOMAIN}}** (o usuário macro {\${DOMAIN}} deve ser definido, preferencialmente no host level)
 - **subpage** : **/release_notes**

* Name

Type

* Key

Type of information

Name	Value	Action
<input type="text" value="url"/>	<input type="text" value="{ \$DOMAIN }"/>	Remove
<input type="text" value="subpage"/>	<input type="text" value="/release_notes"/>	Remove
Add		

* Script

- No campo Script field, digitar:

```
var obj = JSON.parse(value);
var url = obj.url;
var subpage = obj.subpage;
var request = new HttpRequest();
return request.get(url + subpage);
```

Múltiplas solicitações HTTP

Coletar o conteúdo de ambos os endereços <https://www.example.com> https://www.example.com/release_notes:

-Criar um item com o tipo "Script". - No campo Script, digitar:

```
var request = new HttpRequest();
return request.get("https://www.example.com") + request.get("https://www.example.com/release_notes");
```

Fazendo logging

Adicionar a entrada "Log test" no servidor Zabbix and

Adicione a entrada "Log test" ao registro do servidor Zabbix e receba o valor do item "1" como retorno:

- Crie um item com o tipo "Script".
- No campo Script field, digite:

```
Zabbix.log(3, 'Log test');
return 1;
```

1 Criando um item

Visão geral

Para criar um item no frontend do Zabbix, faça o seguinte:

- Vá para: Configuração → Hosts
- Clique em Items na linha do host
- Clique em Create item no canto superior direito da tela
- Insira os parâmetros do item no formulário

Você também pode criar um item abrindo um existente, pressionando o botão Clone e, em seguida, salvando com um nome diferente.

Configuração

A aba **Item** contém atributos genéricos do item.

Item
Tags
Preprocessing

* Name

Load average (5m avg)

Type

Zabbix agent

* Key

system.cpu.load[all,avg5]

Type of information

Numeric (float)

* Host interface

127.0.0.1:10050

Units

* Update interval

1m

Custom intervals

Type	Interval	Period
Flexible	Scheduling	50s
1-7,00:00-24:00		

Add

* History storage period

Do not keep history

Storage period

90d

* Trend storage period

Do not keep trends

Storage period

365d

Value mapping

type here to search

Populates host inventory field

-None-

Description

Enabled

☒

Add


Test


Cancel

Todo os campos de entrada mandatórios são demarcados com um asterico vermelho.

Parâmetro	Descrição
Nome	Nome do item.
Tipo	Tipo do item. Veja sessões individuais de item type .
Chave	Chave to item (até 2048 caracteres). As item keys suportadas podem ser encontradas em seções de tipo de item individual. A chave deve ser única em um host. Se a chave for do tipo 'Zabbix agent', 'Zabbix agent (active)' ou 'Simple check', o valor da chave deve ser suportado pelo Zabbix agent ou Zabbix server. Veja também: o key format correto.

Parâmetro	Descrição
Tipo da informação	<p>Tipo do dado armazenado em uma base de dados após a realização de conversões, se houver.</p> <p>Numérico (sem sinal) - inteiro sem sinal de 64 bits</p> <p>Numérico (ponto flutuante) - número de ponto flutuante de 64 bits</p> <p>Este tipo permitirá a precisão de aproximadamente 15 dígitos em uma faixa de aproximadamente -1.79E+308 to 1.79E+308 (com a exceção do PostgreSQL 11 and earlier versions).</p> <p>Receber valores em notação científica também é suportado. Ex.: 1.23E+7, 1e308, 1.1E-4.</p> <p>Character - dados de texto curto</p> <p>Log - dados de texto longo com propriedades relacionadas ao log opcionais (timestamp, source, severidade, logeventid)</p> <p>Texto - dados de texto longo. Veja também text data limits.</p> <p>Para chaves de itens que retornam dados em apenas um formato específico, o tipo da informação é selecionada de forma automática.</p>
Interface do host	<p>Selecione a interface do host. Este campo está disponível quando estiver editando um item no host.</p>
Unidades	<p>Se o símbolo da unidade for definido, o Zabbix o adiciona após o processamento do valor recebido e o exibirá com o prefixo de unidade definido.</p> <p>Por padrão, se o valor original exceder 1000, ele será dividido por 1000 e exibido de acordo. Por exemplo, se você configurar bps e receber o valor 881764, ele será exibido como 881.76 Kbps.</p> <p>O padrão de memória JEDEC é utilizado para o processar as unidades B (byte), Bps (bytes por segundo), as quais são divididas por 1024. Deste modo, se a unidade for configurada como B ou Bps o Zabbix exibirá:</p> <p>1 como 1B/1Bps</p> <p>1024 como 1KB/1KBps</p> <p>1536 como 1.5KB/1.5KBps</p> <p>O processamento especial é utilizado se as seguintes unidades relacionadas a tempo forem utilizadas:</p> <p>unixtime - traduzido para "aaaa.mm.dd hh:mm:ss". Para traduzir corretamente, o valor recebido deve ser do tipo numérico (sem sinal).</p> <p>uptime - traduzido para "hh:mm:ss" or "N dias, hh:mm:ss"</p> <p>Por exemplo, se você receber um valor como 881764 (segundos), ele será exibido como "10 dias, 04:56:04"</p> <p>s - traduzido para "aaa mmm ddd hhh mmm sss ms"; o parâmetro é tratado como o número de segundos.</p> <p>Por exemplo, se você receber um valor como 881764 (seconds), ele será exibido como "10d 4h 56m"</p> <p>Apenas 3 unidades principais são mostradas, como por exemplo "1m 15d 5h" ou "2h 4m 46s". Caso não tenha dias para serem exibidos, apenas dois níveis serão exibidos - "1m 5h" (minutos, segundos ou milissegundos não são exibidos). Será traduzido para "< 1 ms" se o valor for inferior a 0.001.</p> <p>Observação se o prefixo da unidade for configurado com !, então o prefixo da unidade e/ou o processamento não serão aplicados aos valores do item. Veja unit conversion.</p>

Parâmetro	Descrição
Intervalo de atualização	<p>Coleta um novo valor para este item a cada N segundos. O intervalo máximo de atualização permitido é de 86.400 segundos (1 dia).</p> <p>Time suffixes são suportados, ex.: 30s, 1m, 2h, 1d.</p> <p>User macros são suportadas.</p> <p>Uma única macro deve preencher todo o campo. Não há suporte para várias macros em um campo ou macros misturadas com texto.</p> <p>Observação: o intervalo de atualização pode ser configurado para '0' se os intervalos customizados existirem com um valor diferente de zero. Se for configurado para '0' e o intervalo customizado (flexível ou agendado) existir com um valor diferente de zero, o item será coletado durante a duração do intervalo customizado.</p> <p>Observação: a primeira pesquisa de item após o item se tornar ativo ou após a alteração do intervalo de atualização pode ocorrer antes do valor configurado.</p> <p>Um item passivo existente pode ser coletado para valores imediatamente pressionando o Execute now button.</p>
Intervalos customizados	<p>Você pode criar regras customizadas para a checagem de itens:</p> <p>Flexível - cria uma exceção para o intervalo de atualização (intervalo com frequência distintai)</p> <p>Agendada - cria um esquema de coleta customizado.</p> <p>Para informações mais detalhadas veja Custom intervals.</p> <p>Time suffixes são suportados no campo Intervalo, ex.: 30s, 1m, 2h, 1d.</p> <p>User macros são suportados.</p> <p>Uma única macro deve preencher todo o campo. Não há suporte para várias macros em um campo ou macros misturadas com texto.</p> <p>O agendamento é suportado desde a o Zabbix 3.0.0.</p> <p>Observação: não disponível para itens do tipo agente Zabbix ativo.</p>
Período de armazenamento de histórico	<p>Selecione:</p> <p>Não manter histórico - o histórico do item não será armazenado. Útil para itens do tipo mestre se apenas os itens dependentes devem manter o histórico.</p> <p>Esta configuração não pode ser sobrescrita pelo por settings globais do housekeeper.</p> <p>Período de armazenamentoS - especifica a duração de manter histórico detalhado na base de dados (1 hora a 25 anos). Dados mais antigos serão removidos pelo housekeeper. Armazenado em segundos.</p> <p>Time suffixes são suportados, ex.: 2h, 1d. User macros são suportadas.</p> <p>O valor do Período de armazenamento pode ser sobrescrito globalmente em Administração → Geral → Housekeeper.</p> <p>Se existir uma configuração de sobrescrita global, um ícone verde  será exibido. Se você posicionar o seu cursor sob o ícone, uma mensagem será exibida, ex.: Sobrescrito por configurações globais de limpeza de dados (1d).</p> <p>É recomendado manter os dados coletados pelo menor tempo possível para reduzir o tamanho dos dados históricos na base de dados. Em vez de manter um longo histórico de valores, você pode manter dados mais longos de estatísticas.</p> <p>Veja também History and trends.</p>

Parâmetro	Descrição
Período de armazenamento de estatísticas	<p>Selecione:</p> <p>Não manter dados estatísticos - as estatísticas não serão armazenadas.</p> <p>Esta configuração não poderá ser sobreescrita pela settings global do housekeeper.</p> <p>Período de armazenamento - especifica a duração para manter dados históricos agregados (hourly min, max, avg, count) na base de dados (1 dia a 25 anos). Dados antigos serão removidos pelo housekeeper. Armazenado em segundos.</p> <p>Time suffixes são suportados, ex.: 24h, 1d. User macros são suportados.</p> <p>O valor do Período de armazenamento pode ser sobreescrito globalmente em Administração → Geral → Housekeeper.</p> <p>Se existir uma configuração de sobreescrita global, um ícone verde  será exibido. Se você posicionar o seu cursor sob o ícone, uma mensagem será exibida, ex.: Sobreescrito por configurações globais de limpeza de dados (7d).</p> <p>Observação: Manter estatísticas de dados não numéricos - caracter, log e texto - não está disponível.</p> <p>Veja também History and trends.</p>
Mapeamento de valores	<p>Aplica mapeamento de valores para o item. Value mapping não altera os valores recebidos, é apenas para a exibição de dados. Funciona com itens dos tipo Numérico(sem sinal), Numérico(ponto flutuante) e Caracter.</p> <p>Por exemplo, "Windows service states".</p>
Formato do log de tempo	<p>Disponível somente para itens do tipo Log. Placeholders suportados:</p> <ul style="list-style-type: none"> * y: Ano (1970-2038) * M: Mês (01-12) * d: Dia (01-31) * h: Hora (00-23) * m: Minuto (00-59) * s: Segundo (00-59) <p>Se deixado em branco não haverá o processamento do timestamp. Por exemplo, considere a seguinte linha do arquivo de log do agente Zabbix:</p> <p>" 23480:20100328:154718.045 Zabbix agent started. Zabbix 1.8.2 (revision 11211)."</p> <p>Ela começa com seis posições de caracteres para PID, seguidas por data, hora e o restante da linha.</p> <p>O formato de hora de log para esta linha seria "pppppp:aaaaMMdd:hhmmss".</p> <p>Observe que " p" e ":" são apenas placeholders e podem ser qualquer coisa menos "yMdhms".</p>
Popular campos de inventário do host	<p>Você pode selecionar o campo de inventário do host que aquele valor deve popular. Isso funcionará se o preenchimento automático do inventory estiver configurado para o host.</p> <p>Este campo não está disponível se o Tipo da informação for configurada para 'Log'.</p>
Descrição	<p>Forneça uma descrição do item.</p>
Habilitado	<p>Marque a caixa de seleção para habilitar o item para que seja processado.</p>
Dados recentes	<p>Clique no link para ver os dados mais recentes do item.</p> <p>Este link só está disponível ao editar um item já existente.</p>

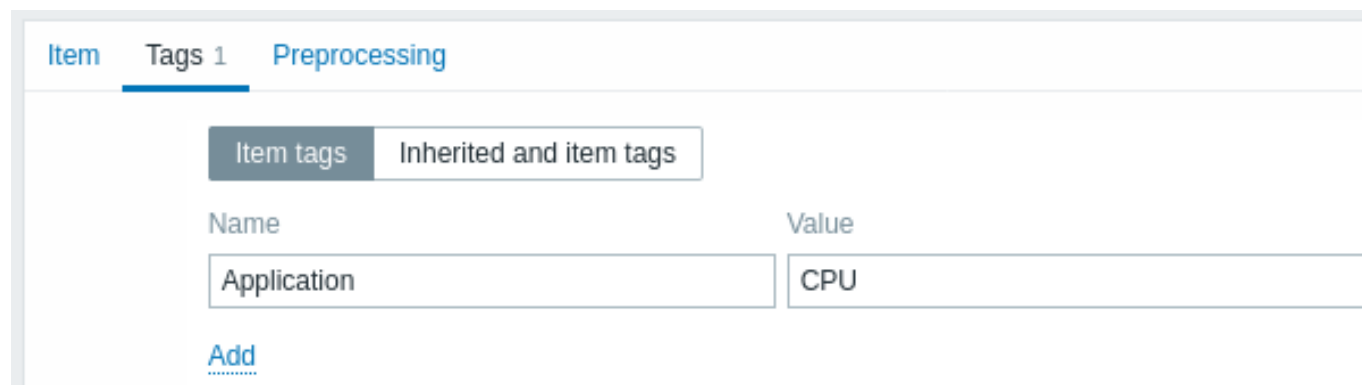
Note:

Campos de tipos de itens específicos estão descritos em [corresponding pages](#).

Note:

Quando estiver editando um item de **template** no host, uma quantidade de campos é de somente leitura. Você pode usar o link no cabeçalho do formulário e ir para a template e editá-los nela, tendo em mente que as alterações em uma template irá alterar o item para todos os hosts em que a template está vinculada.

A aba de **Tags** permite a definição de **tags** de itens.



Name	Value
Application	CPU

Add

Pré-processamento do valor do item

A aba **Pré-processamento** permite definir **transformation rules** para os valores recebidos.

Testando

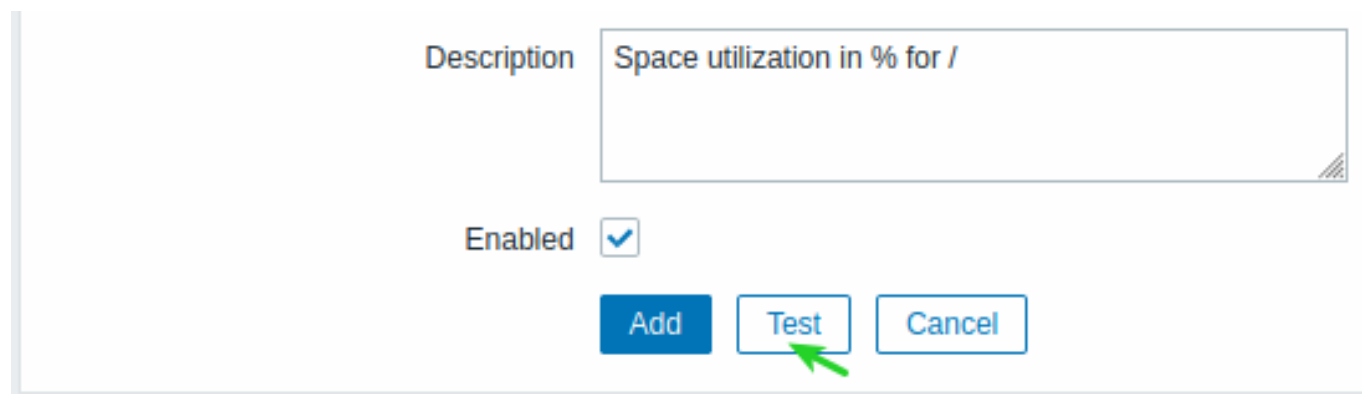
É possível testar um item e, se configurado corretamente, obter um valor de retorno verdadeiro. O teste pode ocorrer antes mesmo do item ser salvo.

O teste é disponível para itens em hosts e templates, protótipos de itens e regras de descoberta. O teste não está disponível para itens do tipo ativo.

O teste de itens está disponível para os seguintes itens do tipo passivo:

- Agent Zabbix
- Agente SNMP (v1, v2, v3)
- Agente IPMI
- Checagem via SSH
- Checagem via Telnet
- Agente JMX
- Checagem simples (exceto itens icmping*, vmware.*)
- Checagem interna do Zabbix
- Itens calculados
- Checagem external
- Monitor de base de dados
- Agente HTTP
- Script

Para testar um item, clique no botão Teste na configuração do item. Note que o botão Teste estará desabilitado se não for possível testar o item (tais como checagens ativas, desconsiderando checagens simples).



Description: Space utilization in % for /

Enabled ☒

Add Test Cancel

O formulário para testar o item possui campos os parâmetros de host necessários (endereço do host, porta, nome do proxy ou sem proxy) e detalhes específicos do item (tais como a comunidade do SNMPv2 ou as credenciais de segurança para o SNMPv3). Esses campos são cientes do contexto:

- Os valores são pré-preenchidos quando possível, ou seja, para itens que requerem agente, pegando as informações da interface do agente selecionado do host
- Os valores devem ser preenchidos manualmente para itens de modelo
- Os valores de macro de texto simples são resolvidos
- Campos onde o valor (ou parte do valor) é um segredo ou Vault macro estão vazias e devem ser inseridas manualmente. Se algum item parâmetro contém um valor de macro secreta, o seguinte aviso mensagem é exibida: "O item contém macros definidas pelo usuário com segredo valores. Os valores dessas macros devem ser inseridos manualmente."
- Os campos são desabilitados quando não são necessários no contexto do item tipo (por exemplo, o campo de endereço do host e o campo de proxy estão desabilitados para itens calculados)

Para testar um item, clique em Obter valor. Se o valor for recebido com sucesso, o campo Valor será preenchido, movendo o valor atual (caso exista) para o campo Valor anterior enquanto calcula o campo Tempo anterior, ex.: a diferença de tempo entre os dois valores e tentando detectar uma sequência de final de arquivo e trocar para CRLF se detectar "\n\r" no valor recebido.

Test item

Get value from host ☒

Host address

192.168.3.205

Port

10050

Proxy

(no proxy)

Get value

Value

33.385793

Time

now

Previous value

Prev. time

End of line sequence

LF

CRLF

Get value and test

Cancel

Se a configuração estiver incorreta, uma mensagem de erro é exibida descrevendo a possível causa.

Test item

Invalid second parameter.

Get value from host ☒

Host address

127.0.0.1

Proxy

(no proxy)

Value

value

A coleta bem sucedida do valor de um host pode ser utilizada para testar os preprocessing steps.

Botões de formulário

Os botões na parte inferior do formulário permitem a realização de diversas operações.

396

Add	Adicione um item. Este botão está disponível apenas para novos itens.
Update	Atualiza as propriedades de um item.
Clone	Cria um outro item baseado nas propriedades do item atual.
Execute now	Executa uma checagem imediata para um novo valor do item. Suportado somente em checagens passivas (veja more details). Observação o verificar um valor imediatamente, o cache de configuração não é atualizado, portanto, o valor não refletirá alterações muito recentes na configuração do item.
Test	Teste se a configuração do item está correta obtendo um valor
Delete and clear	Deleta o histórico e estatísticas de um item.
Delete	Deleta o item.
Cancel	Cancela a edição das propriedades do item.

Limite de dados de texto

O limite de dados de texto depend na base de dados utilizada no backend. Antes de armazenar o texto valores no banco de dados eles são truncados para corresponder ao valor do banco de dados limite de tipo:

Base de dados	Tipo de informação		
	Caracter	Log	Texto
MySQL	255 caracteres	65536 bytes	65536 bytes
PostgreSQL	255 caracteres	65536 caracteres	65536 caracteres
Oracle	255 caracteres	65536 caracteres	65536 caracteres

Conversão de unidade

Por padrão, especificar uma unidade para um item resulta em um prefixo multiplicador sendo adicionado - por exemplo, um valor de entrada '2048' com unidade 'B' ser exibido como '2 KB'.

Para evitar a conversão de uma unidade, use o prefixo !, por exemplo, !B. Para entender melhor como funciona a conversão com e sem o ponto de exclamação, veja os seguintes exemplos de valores e unidades:

```
1024 !B → 1024 B
1024 B → 1 KB
61 !s → 61 s
61 s → 1m 1s
0 !uptime → 0 uptime
0 uptime → 00:00:00
0 !! → 0 !
0 ! → 0
```

Note:

Antes do Zabbix 4.0, havia uma lista de parada de unidade codificada consistindo de ms, rpm, RPM, %. Esta lista de parada foi obsoleto, portanto, a maneira correta de evitar a conversão de tais unidades é !ms, !rpm, !RPM, !%.

Limite de script personalizado

O comprimento do script personalizado disponível depende do banco de dados usado:

Base de dados	Limite em caracteres	Limite em bytes
MySQL	65535	65535
Oracle Database	2048	4000
PostgreSQL	65535	sem limite
SQLite (somente no proxy Zabbix)	65535	sem limite

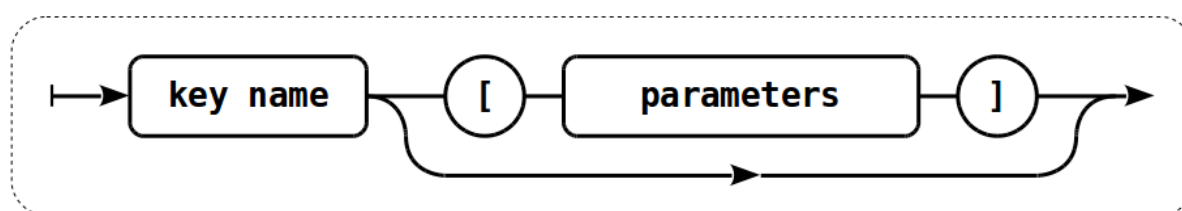
Itens não suportados

Um item pode se tornar não suportado se não for possível coletar o seu valor por algum motivo. Esses itens ainda são verificados novamente em seu **Update interval** padrão.

Itens não suportados são reportados como tendo um estado NÃO SUPORTADO.

1 Formato da chave de item

O formato da chave do item, incluindo parâmetros da chave, deve seguir as regras de sintaxe. As seguintes ilustrações mostram a sintaxe suportada. Elementos e caracteres permitidos em cada ponto podem ser determinados seguindo as setas - se algum bloco pode ser alcançado pela linha, ele é permitido, se não - ele não é permitido.



Para construir uma chave de item válida, se começa especificando o nome da chave, então há a escolha de ter parâmetros ou não - como representado pelas duas linhas que poderiam ser seguidas.

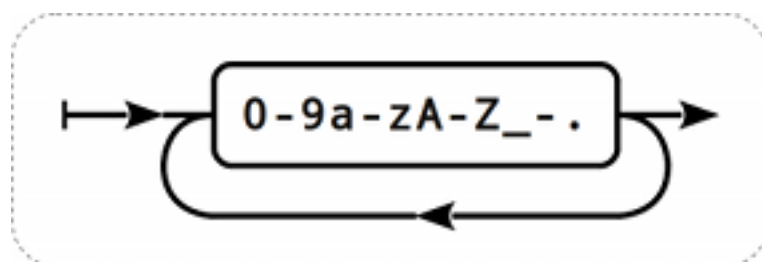
Nome da chave

O nome da chave tem um intervalo limitado de caracteres permitidos, que se seguem. Os caracteres permitidos são:

0-9a-zA-Z_ - .

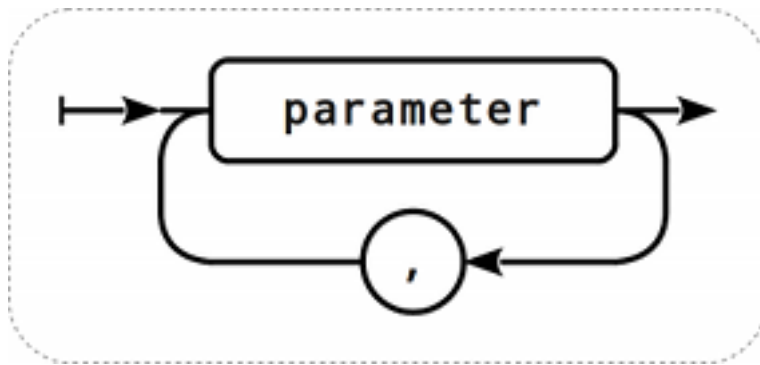
O que significa:

- todos os números;
- todas as letras minúsculas;
- todas as letras maiúsculas;
- sublinhado;
- traço;
- ponto.

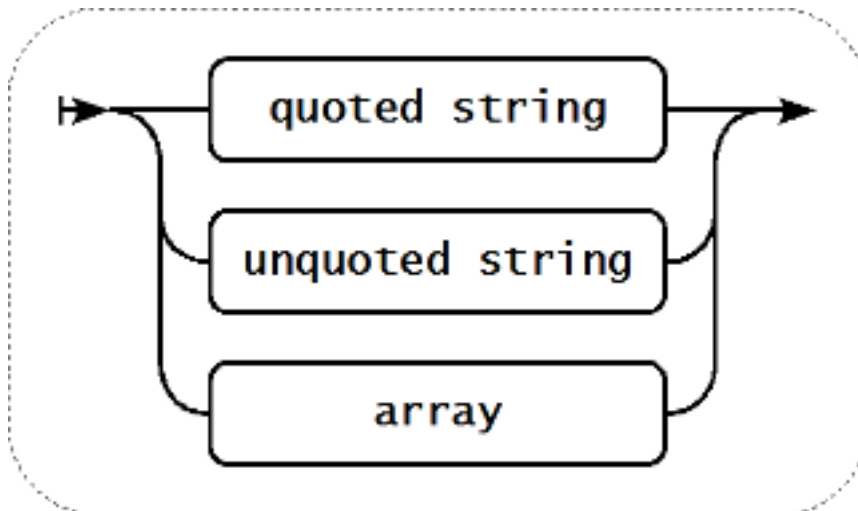


Parâmetros de chave

Uma chave de item pode ter múltiplos parâmetros que são separados por vírgula..



Cada parâmetro da chave pode ser ou um texto com aspas, um texto sem aspas ou um array.



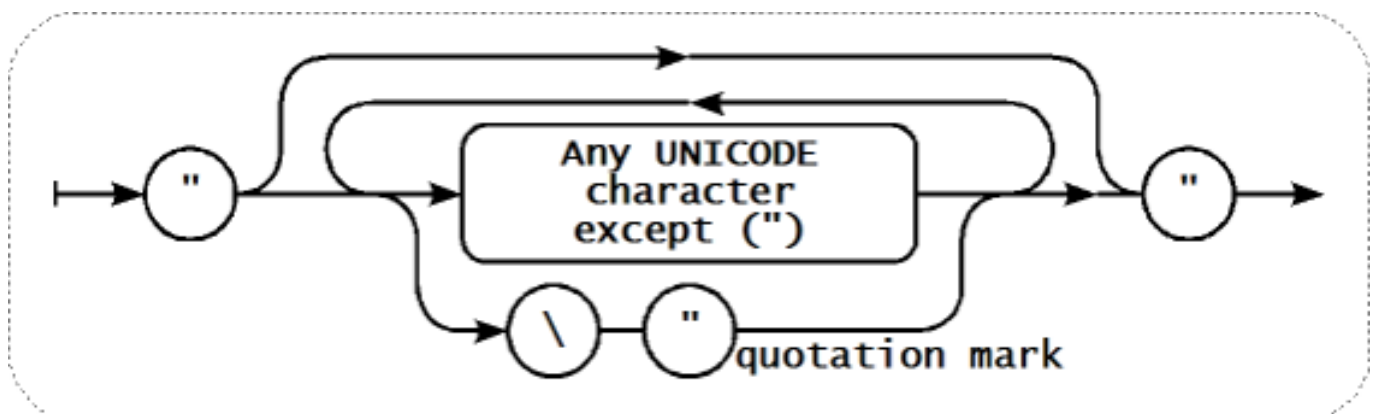
O parâmetro também pode ser deixado vazio, usando assim o valor padrão. Deste modo, o número apropriado de vírgulas deve ser informado se quaisquer parâmetros adicionais forem especificados. Por exemplo, a chave **icmpping[„200„500]** especificaria que o intervalo entre cada ping é de 200 milissegundos, tempo máximo (timeout) de 500 milissegundos, e todos os outros parâmetros continuam com seus valores padrão.

Parâmetro - texto com aspas (quotado)

Se o parâmetro da chave é um texto quotado, qualquer caracter Unicode é permitido.

Se o parâmetro contém vírgula, este parâmetro tem que ser quotado.

Se o texto do parâmetro contém aspas, este parâmetro tem que ser quotado e todas as aspas que fazem parte do parâmetro têm que ser escapadas com um caracter de contrabarra (\).

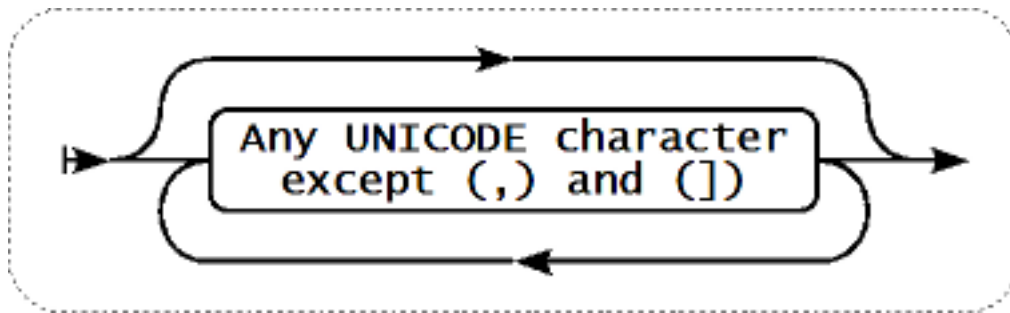


Warning:

Para quotar os parâmetros da chave, use apenas aspas duplas. Aspas simples não são suportadas.

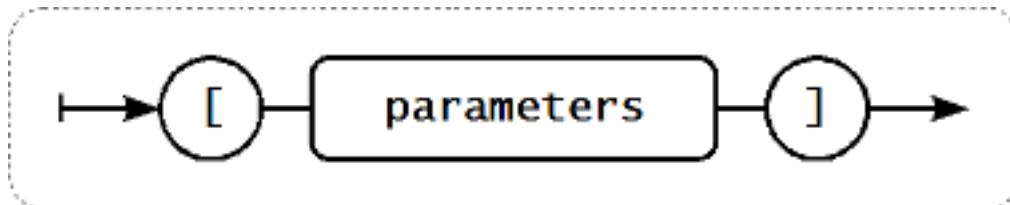
Parâmetro - texto sem aspas (não quotado)

Se o parâmetro da chave é um texto não quotado, qualquer caracter Unicode é permitido, exceto vírgula e colchete final (]). Um parâmetro não quotado não pode começar com colchete inicial ([).



Parâmetro - array

Se o parâmetro chave for um array, ele é novamente incluído entre colchetes, onde parâmetros individuais seguem as regras e a sintaxe de especificação de múltiplos parâmetros.



Attention:

Arrays de parâmetros de múltiplos níveis, por exemplo, [a, [b, [c,d]], e], não são permitidos

2 Intervalos customizados

Visão geral

É possível criar regras customizadas em relação ao tempo em que um item é verificado. Os dois métodos disponíveis são Intervalos flexíveis, que permitem redefinir o intervalo de atualização padrão, e Agendamento, pelo qual uma verificação pode ser executada em um horário específico ou sequência de horários.

Intervalos flexíveis

Intervalos flexíveis permitem redefinir o intervalo de atualização padrão para períodos de tempo específicos. Um intervalo flexível é definido com Intervalo e Período onde:

- Intervalo – o intervalo de atualização para o período de tempo especificado
- Período – o período de tempo quando o intervalo flexível está ativo (consulte [períodos de tempo](#) para uma descrição detalhada do formato para Período)

Até sete intervalos flexíveis podem ser definidos. Se múltiplos intervalos se sobrepõem, o menor valor de intervalo é utilizado para o período de sobreposição. Note que se o menor valor de sobreposição de intervalos for '0', não haverá verificação passiva (polling). Fora dos intervalos flexíveis é usado o intervalo de atualização padrão.

Note que se o intervalo flexível se iguala à largura do período, o item será verificado exatamente uma vez. Se o intervalo é maior que o período, o item pode ser verificado uma vez ou ele pode nem ser verificado (por isso tal configuração não é aconselhável). Se o intervalo flexível é menor que o período, o item será verificado ao menos uma vez.

Se o intervalo flexível for configurado para '0', o item não é verificado durante o período do intervalo flexível e retorna a verificação de acordo com o Intervalo de atualização padrão assim que o período tiver findado. Exemplos:

Intervalo	Período	Descrição
10	1-5,09:00-18:00	O item será verificado a cada 10 segundos durante o horário de trabalho.
0	1-7,00:00-7:00	O item não será verificado durante a madrugada.
0	7-7,00:00-24:00	O item não será verificado aos domingos.
60	1-7,12:00-12:01	O item será verificado todos os dias ao meio-dia. Note que isto foi usado como contingência para verificações agendadas e a partir do Zabbix 3.0 é recomendado usar intervalos de agendamento para tais verificações.

Intervalos de agendamento

Intervalos de agendamento são utilizados para verificar itens em horários específicos. Enquanto os intervalos flexíveis são projetados para redefinir o intervalo de atualização padrão do item, os intervalos de agendamento são usados para especificar uma agenda de verificação independente, que é executada de forma paralela.

Um intervalo de agendamento é definido como: `md<filtro>wd<filtro>h<filtro>m<filtro>s<filtro>` onde:

- **md** - mês dias
- **wd** - semana dias
- **h** - horas
- **m** - minutos
- **s** - segundos

O `<filtro>` é usado para especificar valores para seu prefixo (dias, horas, minutos, segundos) e é definido como: `[<de>[-<até>]][/<salto>][,<filtro>]` onde:

- `<de>` e `<até>` definem o intervalo de valores considerados (inclusive eles próprios). Se `<até>` for omitido então o filtro interpreta o intervalo como `<de>` - `<de>`. Se `<de>` também for omitido então o filtro considerará como intervalo todos os valores possíveis.
- `<salto>` define o salto dentro do intervalo. Por padrão `<salto>` tem o valor 1, o que significa que todos os valores do intervalo definido são considerados.

Mesmo sendo as definições de filtros opcionais, ao menos um filtro deve ser usado. Um filtro deve definir um intervalo ou um valor de `<salto>`.

Um filtro vazio corresponde a '0' se não houver definição de filtros nos níveis abaixo, caso contrário, todos os valores possíveis serão considerados. Por exemplo, se o filtro de hora é omitido então apenas a hora '0' será considerada, contanto que os filtros de minutos e segundos também tenham sido omitidos, caso contrário o filtro de hora vazio considerará todos os valores de hora possíveis.

Valores válidos para `<de>` e `<até>` em seus respectivos prefixos são:

Prefixo	Descrição	<de>	<até>
md	Mês dias	1-31	1-31
wd	Semana dias	1-7	1-7
h	Horas	0-23	0-23
m	Minutos	0-59	0-59
s	Segundos	0-59	0-59

O valor `<de>` deve ser menor ou igual ao valor de `<até>`. O `<salto>` deve ser maior ou igual a 1 e menor ou igual a `<até>` - `<de>`.

Valores simples para dias do mês, horas, minutos e segundos podem ser prefixados com 0. Por exemplo, `md01-31` e `h/02` são intervalos válidos, mas `md01-031` e `wd01-07` não são.

No Zabbix Frontend, múltiplos intervalos de agendamento são informados em linhas separadas. No Zabbix API, eles são concatenados em um texto único com ponto-e-vírgula ; como separador.

Se um tempo é correspondido por vários intervalos ele é executado apenas uma vez. Por exemplo, `wd1h9;h9` será executado apenas uma vez na segunda às 9h.

Exemplos:

Intervalo	Será executado
m0-59	a cada minuto
h9-17/2	a cada 2 horas começando às 9h (9h, 11h ...)
m0,30 or m/30	de hora em horas às Xh e Xh30
m0,5,10,15,20,25,30,35,40,45,50,55 or m/5	a cada cinco minutos
wd1-5h9	toda segunda a sexta às 9h
wd1-5h9-18	toda segunda a sexta às 9h,10h,...,18h
h9,10,11 or h9-11	todo dia às 9h, 10h e 11h
md1h9m30	todo 1º dia de cada mês às 9h30
md1wd1h9m30	todo 1º dia de cada mês às 9h30 se for segunda
h9m/30	todo dia às 9h, 9h30
h9m0-59/30	todo dia às 9h, 9h30

Intervalo	Será executado
h9,10m/30	todo dia às 9h, 9h30, 10h, 10h30
h9-10m30	todo dia às 9h30, 10h30
h9m10-40/30	todo dia às 9h10, 9h40
h9,10m10-40/30	todo dia às 9h10, 9h40, 10h10, 10h40
h9-10m10-40/30	todo dia às 9h10, 9h40, 10h10, 10h40
h9m10-40	todo dia às 9h10, 9h11, 9h12, ... 9h40
h9m10-40/1	todo dia às 9h10, 9h11, 9h12, ... 9h40
h9-12,15	todo dia às 9h, 10h, 11h, 12h, 15h
h9-12,15m0	todo dia às 9h, 10h, 11h, 12h, 15h
h9-12,15m0s30	todo dia às 9h0m30s, 10h0m30s, 11h0m30s, 12h0m30s, 15h0m30s
h9-12s30	todo dia às 9h0m30s, 9h01m30s, 9h02m30s ... 12h58m30s, 12h59m30s
h9m/30;h10 (Sintaxe específica da API)	todo dia às 9h, 9h30, 10h
h9m/30	todo dia às 9h, 9h30, 10h
h10 (adicione este como outra linha no Frontend)	

Aligning time zones for proxies and agent 2

Note that Zabbix proxies and agent 2 use their local time zones when processing scheduling intervals.

For this reason, when scheduling intervals are applied to items monitored by Zabbix proxy or agent 2 active items, it is recommended to set the time zone of the respective proxies or agent 2 the same as Zabbix server, otherwise the **queue** may report item delays incorrectly.

The time zone for Zabbix proxy or agent 2 can be set using the environment variable TZ in the systemd unit file:

```
[Service]
...
Environment="TZ=Europe/Amsterdam"
```

2 Pré-processamento de valor de item

Visão geral

O pré-processamento permite definir regras de transformação para os valores recebidos nos itens. Uma ou várias transformações são possíveis antes de salvar a informação no banco de dados.

Transformações são executadas na ordem em que são definidas. O pré-processamento é feito pelo Zabbix Server ou Proxy (se os itens são monitorados pelo Proxy).

Note que a conversão para o tipo de valor desejado (como definido na configuração do item) é executada no fim da série de pré-processamento; as conversões, no entanto, também podem ocorrer caso requerido pela etapa de pré-processamento correspondente. Consulte [detalhes de pré-processamento](#) para mais informações técnicas.

Veja também: [Exemplos de uso](#)

Configuração

Regras de pré-processamento são definidas na aba de **Pré-processamento** na tela de [configuração](#) do item.

Items

All hosts / Zabbix server Enabled ZBX Items 145 Triggers 75 Graphs 28 Discovery rules 3 Web scenarios

Item Tags Preprocessing 2

Preprocessing steps	Name	Parameters	Custom on fail
1:	Change per second		<input type="checkbox"/>
2:	Custom multiplier	0.01	<input type="checkbox"/>

Add

Type of information Numeric (float)

Add Test Cancel

Attention:

Um item se tornará **não suportado** se qualquer uma das etapas de pré-processamento falhar, a menos que tenha sido especificado um tratamento de erro customizado usando a opção Personalizado em caso de falha para transformações suportadas.

Para itens de log, os metadados de log (sem valor) sempre atualizarão o estado do item não suportado, tornando-o novamente suportado, mesmo que o erro inicial tenha ocorrido após o recebimento do valor do log a partir do Agent.

Macros de usuário e macros de usuário com contexto são suportadas nos parâmetros de pré-processamento de valores dos itens, incluindo código JavaScript.

Note:

O contexto é ignorado quando uma macro é substituída por seu valor. O valor da macro é inserido no código tal como é, não sendo possível inserir caracteres de escape adicionais após a gravação do valor no código JavaScript. Por favor, esteja ciente que este comportamento pode causar erros de JavaScript em alguns casos.

Tipo		
	Transformação	Descrição
Texto		

Expressão regular

Combina o valor com o <padrao> da expressão regular e substitui o valor com a <saida>. A expressão regular suporta a extração de no máximo 10 grupos capturados com a sequência \N. Falha em combinar o valor de entrada tornará o item não suportado.

Parâmetros:

padrão - expressão regular

saída - modelo de formatação de saída. Uma sequência de escape \N (onde N=1...9) é substituída pelo N-ésimo grupo correspondente. Uma sequência de escape \0 é substituída pelo texto verificado. Por favor, consulte a seção **expressões regulares** para alguns exemplos disponíveis. Se você ativar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Substituir	<p>Encontre o texto buscado e substitua por outro (ou nada). Todas as ocorrências do texto buscado serão substituídas. Parâmetros:</p> <p>texto buscado - o texto para encontrar e substituir, sensível à maiúsculas e minúsculas (obrigatório)</p> <p>substituição - o texto com o qual substituir o valor buscado. O texto de substituição também pode ser vazio, permitindo apagar o texto buscado quando encontrado. É possível utilizar sequências de escape para buscar ou substituir quebras de página, caracter de retorno, tabulações e espaços "\n \r \t \s"; contrabarra pode ser escapada como "\\" e sequências de escape podem ser escapadas como "\\n". O escape de quebras de linha, caracter de retorno e tabulação é feito automaticamente durante descobertas de baixo-nível.</p>
Remover	<p>Remove os caracteres especificados do início e fim do valor.</p>
Remover à direita	<p>Remove os caracteres especificados do fim do valor.</p>
Remover à esquerda	<p>Remove os caracteres especificados do início do valor.</p>

XML XPath

Extraia um valor ou fragmento de dados XML usando a funcionalidade XPath.

Para esta opção funcionar, o Zabbix Server deve ser compilado com suporte a libxml.

Exemplos:

`number(/document/item/value)`
irá extrair 10 de
`<document><item><value>10`
`number(/document/item/@attribute="10")`
irá extrair 10 de
`<document><item attribute="10"></item></document>`
irá extrair
`<item><value>10</value></item>`
de
`<document><item><value>10`

Note que namespaces não são suportados.

Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

JSON Path	<p>Extraia um valor ou fragmento de dados JSON usando a funcionalidade JSONPath.</p> <p>Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.</p>
CSV para JSON	<p>Converte dados CSV para o formato JSON.</p> <p>Para mais informações, veja: pré-processamento CSV para JSON.</p>
XML para JSON	<p>Converte dados no formato XML para JSON.</p> <p>Para mais informações, veja: Regras de serialização.</p> <p>Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.</p>

Tipo

Aritmética

Multiplicador
customizado

Multiplica o valor pelo número inteiro ou ponto flutuante especificado. Use esta opção para converter valores recebidos em KB, MBps, etc. em B, Bps. Caso contrário o Zabbix não consegue configurar os **prefixos** (K, M, G, etc.) corretamente. Note que se o tipo de informação do item é Numérico (unsigned), valores de entrada com uma parte fracionada serão encurtados (p.e. '0.9' se tornará '0') antes que o multiplicador customizado seja aplicado. Suportado: notação científica, por exemplo, 1e+70 (desde a versão 2.2); macros de usuário e macros LLD (desde a versão 4.0); textos que incluem macros, por exemplo, {#MACRO}e+10, {\$MACRO1}e+{\$MACRO2} (desde a versão 5.2.3) As macros devem resolver para um número inteiro ou ponto flutuante. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar

Tipo

Mudança

Mudança simples

Calcula a diferença entre os valores atual e anterior. Calculado como **valor - valor_anterior**, onde **valor** - valor atual; **valor_anterior** - valor recebido anteriormente. Esta opção pode ser útil para medir um valor em constante crescimento. Se o valor atual é menor que o valor recebido anteriormente, o Zabbix descarta essa diferença (não armazena nada) e aguarda um novo valor. Apenas uma operação de mudança por item é permitida. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Mudança por segundo

Calcula a mudança de valor (diferença entre os valores atual e anterior) em velocidade por segundo. Calculado como **(valor - valor_anterior)/(tempo - tempo_anterior)**, onde
 valor - valor atual;
 valor_anterior - valor recebido anteriormente;
 tempo - registro de data e hora atual;
 tempo_anterior - registro de data e hora do valor recebido anteriormente.
 Esta opção é extremamente útil para obter a velocidade por segundo de um valor em constante crescimento. Se o valor atual for menor que o valor anterior, o Zabbix descarta a diferença (não armazena nada) e aguarda por outro valor. Isto ajuda a trabalhar corretamente com, por exemplo, encapsulamento (estouro) de contadores SNMP 32-bit.
 Nota: Como este cálculo pode produzir números de ponto flutuante, é recomendado configurar o 'Tipo de informação' para Numérico (float), mesmo que os números brutos de entrada sejam inteiros. Isto é especialmente relevante para números pequenos onde a parte decimal é importante. Se os valores de ponto flutuante são

Booleano para decimal

Converte um valor do formato booleano para decimal. A representação textual é traduzida para 0 ou 1. Assim, 'TRUE' é armazenado como 1 e 'FALSE' é armazenado como 0. Todos os valores são avaliados de forma não sensível a maiúsculas e minúsculas. Atualmente os valores reconhecidos são, para:
TRUE - true, t, yes, y, on, up, running, enabled, available, ok, master
FALSE - false, f, no, n, off, down, unused, disabled, unavailable, err, slave
Adicionalmente, qualquer valor numérico diferente de zero é considerado ser TRUE e zero é considerado ser FALSE.
Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Octal para decimal

Converte um valor do formato octal para decimal. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Hexadecimal para decimal

Converte um valor do formato hexadecimal para decimal. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Scripts personalizados

Tipo	
Validação	<div>JavaScript</div> <div> <p>Insira o código JavaScript no bloco que aparece quando clicando no campo parâmetro ou sobre o ícone de edição (lápis). Note que o tamanho disponível para o código JavaScript depende do banco de dados usado. Para mais informações, veja: Pré-processamento Javascript.</p> </div>

Dentro do intervalo

Define um intervalo dentro do qual um valor deve estar pela definição dos valores mínimo/máximo (inclusive). Valores numéricos são aceitos (incluindo qualquer quantidade de dígitos, partes decimal e exponencial opcionais e valores negativos). Macros de usuário e descoberta de baixo-nível podem ser utilizadas. O valor mínimo deve ser menor que o valor máximo. Ao menos um valor deve existir. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Corresponde com expressão regular

Especifique uma expressão regular com a qual um valor deve corresponder. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Não corresponde com expressão regular

Especifique uma expressão regular com a qual um valor não deve corresponder. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Verifica erro em
JSON

Verifica mensagem de erro no nível de aplicação localizada no JSONpath. Encerra o processamento se tiver sucesso e a mensagem não estiver vazia; caso contrário, continua processando com o valor existente antes desta etapa de pré-processamento. Note que estes erros de serviços externos são reportados ao usuário como são, sem adicionar informações da etapa de pré-processamento. Nenhum erro será reportado em caso de falha ao analisar um JSON inválido. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Verifica erro em XML

Verifica mensagem de erro no nível de aplicação localizada no XPath. Encerra o processamento se tiver sucesso e a mensagem não estiver vazia; caso contrário, continua processando com o valor existente antes desta etapa de pré-processamento. Note que estes erros de serviços externos são reportados ao usuário como são, sem adicionar informações da etapa de pré-processamento. Nenhum erro será reportado em caso de falha ao analisar o XML. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Verifica erro
usando expressão
regular

Verifica mensagem de erro no nível de aplicação usando expressão regular. Encerra o processamento se tiver sucesso e a mensagem não estiver vazia; caso contrário, continua processando com o valor existente antes desta etapa de pré-processamento. Note que estes erros de serviços externos são reportados ao usuário como são, sem adicionar informações da etapa de pré-processamento. Parâmetros:
padrão - expressão regular
saída - modelo de formatação de saída. Uma sequência de escape \N (onde N=1...9) é substituída pelo N-ésimo grupo correspondente. Uma sequência de escape \0 é substituída pelo texto verificadot. Se você marcar a opção Personalizado em caso de falha, o item não se tornará não suportado no caso de falha em uma etapa de pré-processamento e é possível especificar opções de tratamento de erro personalizadas: seja para descartar o valor, configurar um valor específico ou configurar uma mensagem de erro específica.

Verifica valor não suportado

Verifica se há erro na recepção do valor de um item. Normalmente isto faria o item se tornar não suportado, mas você pode modificar este comportamento configurando as opções de tratamento Personalizado em caso de falha: seja para descartar o valor, configurar um valor específico (neste caso o item continua suportado e o valor pode ser usado nos gatilhos (triggers)) ou configurar uma mensagem de erro específica. Note que para esta etapa de pré-processamento, a opção Personalizado em caso de falha aparecerá em cinza e sempre selecionada. Esta etapa é sempre executada como a primeira etapa de pré-processamento e é colocada acima de todas as outras após salvar as alterações no item. Ela pode ser usada apenas uma vez. Suportado desde a versão 5.2.0.

Supressão

Descartar
inalterado

Descarta um valor caso ele não tenha mudado.
Se um valor é descartado, ele não é salvo no banco de dados e o Zabbix Server não tem conhecimento de que este valor foi recebido.
Nenhum gatilho será avaliado, como resultado, nenhum problema será criado/resolvido para os gatilhos relacionados.
Funções funcionarão baseadas apenas nos dados salvos no banco de dados.
Como tendências são contruídas baseadas nos dados do banco de dados, se não há valor salvo por uma hora então também não haverá dados de tendência para aquela hora.
Apenas uma opção de supressão pode ser especificada para um item.
Note que é possível para itens monitorados pelo Zabbix Proxy que diferenças de valores muito pequenas (menos de 0.000001) sejam corretamente não descartadas pelo Proxy, mas armazenadas no histórico como sendo de mesmo valor se o banco de dados do Zabbix Server não tiver sido atualizado.

Descartar
inalterado com
intervalo

Descarta um valor caso ele não tenha mudado dentro de um período de tempo definido (em segundos). Valores inteiros positivos são suportados para especificar os segundos (mínimo - 1 segundo). Sufixos de tempo podem ser usados neste campo (p.e. 30s, 1m, 2h, 1d). Macros de usuário e descoberta de baixo-nível podem ser usadas neste campo. Se um valor é descartado, ele não é salvo no banco de dados e o Zabbix Server não tem conhecimento de que este valor foi recebido. Nenhum gatilho será avaliado, como resultado, nenhum problema será criado/resolvido para os gatilhos relacionados. Funções funcionarão baseadas apenas nos dados salvos no banco de dados. Como tendências são contruídas baseadas nos dados do banco de dados, se não há valor salvo por uma hora então também não haverá dados de tendência para aquela hora. Apenas uma opção de supressão pode ser especificada para um item. Note que é possível para itens monitorados pelo Zabbix Proxy que diferenças de valores muito pequenas (menos de 0.000001)

Tipo		
Prometheus	Padrão Prometheus	Use a seguinte instrução para extrair os dados necessários das métricas do Prometheus. Consulte verificações Prometheus para mais detalhes.
	Prometheus para JSON	Converte métricas Prometheus para JSON. Consulte verificações Prometheus para mais detalhes.

Attention:

Para as etapas de pré-processamento de alteração e supressão o Zabbix tem que se recordar do último valor para calcular/comparar com o novo valor conforme necessidade. Estes últimos valores são tratados pelo gerenciador de pré-processamento. Se o Zabbix Server ou Proxy for reiniciado ou houver qualquer mudança nas etapas de pré-processamento, o último valor do item correspondente é resetado, resultando em:

- para as etapas Alterações simples, Alterações por segundo - o próximo valor será ignorado pois não há valor anterior para cálculo da alteração;
- para as etapas Descartar inalterado, Descartar inalterado com intervalo - o próximo valor nunca será descartado, mesmo que tivesse de ser descartado por causa das regras de descarte.

O parâmetro Tipo de informação é apresentado no fundo da página quando ao menos uma etapa de pré-processamento for definida. Se necessário, é possível alterar o tipo de informação sem sair da tela de Pré-processamento. Consulte [Criando um item](#) para uma descrição detalhada do parâmetro.

Note:

Se você usa um multiplicador customizado ou armazena valores como Alterações por segundo para itens com o tipo de informação configurado para Numérico (unsigned) e o valor calculado resultante for na verdade um número de ponto flutuante, o valor calculado ainda é aceito como um valor correto pela retirada da parte decimal e armazenamento do valor como um inteiro.

Testes

O teste das etapas de pré-processamento é útil para certificar-se de que segmentos complexos de pré-processamento entreguem os resultados que são esperados, sem aguardar que o valor do item seja recebido e pré-processado.

Item
Tags
Preprocessing 3

Preprocessing steps

	Name	Parameter	Custom on fail	Actions
1:	Regular expression	<div> <div> <div> <div> </div> </div> <div> </div> </div> </div>		

 | [Test](#) [Re...](#) || 2: | Regular expression | |

 | [Test](#) [Re...](#) || 3: | Regular expression | |

 | [Test](#) [Re...](#) || [Add](#) | | | | [Test all s...](#) |

Type of information

Text

Add

Test

Cancel


É possível testar:

- contra um valor hipotético
- contra um valor real de um host

Cada etapa de pré-processamento pode ser testada individualmente assim como todas as etapas podem ser testadas juntas. Quando você clica no botão Testar ou Testar todas as etapas na aba de Ações, uma janela de teste é aberta.

Testando valor hipotético

Test item

 cannot perform regular expression "[0-9+]" match for value of type "string": invalid regular expression: missing terminating] for character class

Get value from host ☐

Value

Time

☐ Not supported


Previous value

Prev. time

End of line sequence

☒ LF ☐ CRLF

Preprocessing steps

Name	Result
1: Regular expression	15
2: Regular expression	1
3: Regular expression	

Test

Cancel

Parâmetro	Descrição
Obter valor do host	Se você deseja testar um valor hipotético, deixe esta opção desmarcada.
Valor	Veja também: Testando valor real . Informe um valor de entrada para testar. Clicando no campo deste parâmetro ou no botão de visualizar/editar abrirá uma janela com área de texto para entrada do valor ou bloco de código.
Não suportado	Marque esta opção para testar um valor não suportado. Esta opção é útil para testar a etapa de pré-processamento Verificar valores não suportados.
Horário	Horário do valor de entrada é mostrado: now (somente leitura), sendo 'now' o horário atual.
Valor anterior	Informe um valor de entrada anterior para fazer comparação. Apenas para etapas de pré-processamento Alteração e Supressão.
Horário anterior	Informe um valor de horário anterior para fazer comparação. Apenas para etapas de pré-processamento Alteração e Supressão. O valor padrão é baseado no campo de 'Intervalo de atualização' do item (se '1m', então este campo é preenchido com now-1m). Se nada for especificado ou se o usuário não tiver acesso ao host, o padrão é now-30s.
Macros	Se quaisquer macros estiverem sendo usadas, elas são listadas junto de seus valores. Os valores são editáveis para propósitos de teste, mas as alterações serão salvas apenas dentro do contexto de teste.
Sequência de fim de linha	Selecione a sequência de fim de linha para múltiplos valores de entrada: LF (line feed) - sequência de alimentação de linha (comum Linux) CRLF (carriage-return line-feed) - sequência de alimentação de linha (comum Windows).

424

Parâmetro	Descrição
Etapas de pré-processamento	Etapas de pré-processamento são listadas; o resultado do teste é mostrado para cada etapa depois que o botão Testar é acionado. Se uma etapa falhar durante o teste, um ícone de erro é mostrado. A descrição do erro é apresentada ao mover o mouse sobre o ícone. No caso de "Personalização em caso de falha" estar especificada para a etapa e tal ação for executada, uma nova linha aparecerá logo após a linha de teste da etapa de pré-processamento, mostrando qual ação foi executada e qual resultado foi produzido (erro ou valor).
Resultado	O resultado final do teste das etapas de pré-processamento é mostrado em todos os casos quando todas as etapas são testadas juntas (quando você clica em Testar todas as etapas). O tipo de conversão para o tipo de valor do item também é mostrado, por exemplo Resultado convertido para Numérico (unsigned).

Clique em Testar para ver o resultado junto de cada etapa de pré-processamento.

Os valores de teste são armazenados entre as sessões de teste para cada etapa individual ou todas as etapas, permitindo ao usuário alterar as etapas de pré-processamento ou configuração de item e então retornar para a janela de teste sem ter que reinserir informações. Estes valores são perdidos, no entanto, caso a página seja atualizada.

O teste é executado pelo Zabbix Server. O Frontend envia uma requisição correspondente para o Server e aguarda pelo resultado. A requisição contém a entrada de valor e etapas de pré-processamento (com macros de usuário expandidas). Para etapas de Alteração e Supressão, um valor anterior e horário opcionais podem ser especificados. O Server responde com os resultados para cada etapa de pré-processamento.

Todos os erros técnicos ou erros de validação de entrada são mostrados na caixa de erro no topo da janela de teste.

Testando valor real

Para testar o pré-processamento contra um valor real:

- Marque a opção Obter valor do host
- Informe ou verifique os parâmetros do host (endereço de host, porta, nome do proxy/sem proxy) e detalhes específicos do item (como comunidade SNMPv2 ou credenciais de segurança SNMPv3). Estes campos são sensíveis ao contexto:
 - Os valores são pré-preenchidos quando possível, p.e. para itens que requerem um Agent, pela aquisição da informação a partir da interface de Agent do host selecionada
 - Os valores tem de ser preenchidos manualmente para modelos de itens
 - Valores de macro em texto-plano são resolvidos
 - Campos onde o valor (ou parte do valor) seja secreto ou uma macro Vault aparecem vazios e devem ser informados manualmente. Se algum parâmetro de item contém um valor de macro secreto, a seguinte mensagem de alerta é mostrada: "O item contém macros definidas pelo usuário com valores secretos. Os valores para estas macros devem ser informados manualmente."
 - Os campos ficam desabilitados quando desnecessários no contexto do tipo de item (p.e. o endereço do host e campos de Proxy ficam desabilitados para itens calculados)
- Clique em Obter valor e testar para testar o pré-processamento

Test item

Get value from host

* Host address

127.0.0.1

Port

10050

Proxy

(no proxy)

Value

5.4.0alpha1

Time

now

Previous value

5.4.0alpha1

Prev. time

now-7s

End of line sequence

LF

CRLF

Preprocessing steps

Name

1: Discard unchanged with heartbeat

Result

No value

Result

Result converted to Character

No value

Get value and test

Cancel

Se você especificou um mapeamento de valor no formulário de configuração do item (campo 'Mostrar valor'), o diálogo de teste do item mostrará uma outra linha após o resultado final, identificada como 'Resultado com mapeamento de valor aplicado'.

Parâmetros que são específicos para obtenção de um valor real de um host:

Parâmetro	Descrição
Obter valor do host	Marque esta opção para obter um valor real do host.
Endereço do Host	Informe o endereço do host. Este endereço é automaticamente preenchido pelo endereço da interface de host do item.
Porta	Informe a porta do host. Este endereço é automaticamente preenchido pela porta da interface de host do item.
Campos adicionais para interfaces SNMP (Versão SNMP, Comunidade SNMP, Nome de contexto, etc.)	Veja Configurando monitoramento SNMP para detalhes adicionais na configuração de uma interface SNMP (v1, v2 and v3). Estes campos são automaticamente preenchidos a partir da interface de host do item.
Proxy	Especifique o Proxy se o host for monitorado por um Proxy. Este campo é automaticamente preenchido pelo Proxy do host (se houver).

Para os demais parâmetros, veja [Testando valores hipotéticos](#) acima.

1 Exemplos de uso

Visão geral

Esta seção fornece exemplos de uso das etapas de pré-processamento para realização de algumas tarefas práticas.

Filtro de registros de log de eventos VMware

Usando um pré-processamento de expressão regular para filtrar eventos desnecessários do log de eventos VMWare.

1. Em um host VMWare Hypervisor funcional verifique que o item de log de evento `vmware.eventlog[<url>,<mode>]` esteja presente e funcionando de forma apropriada. Note que o item de log de evento pode já estar presente no hypervisor se o modelo Modelo VM VMWare foi associado durante a criação do host.

2. No host VMWare Hypervisor crie um **item dependente** do tipo 'Log' e defina o item de log de evento como principal.

Na aba de "Pré-processamento" do item dependente selecione a opção de validação "Corresponde à expressão regular" e preencha com o padrão, por exemplo:

"* logged in *" - filtra todos os eventos de log no log de eventos

"\bUser\s+\K\S+" - filtra apenas linhas com nomes de usuário do log de eventos

Attention:

Se a expressão regular não é correspondida então o item dependente se torna não suportado com uma mensagem de erro correspondente. Para evitar isto, marque a caixa de seleção "Customizado em falha" e selecione o descarte do valor não correspondido, por exemplo.

Outra abordagem que permite o uso de grupos de correspondência e controle de saída é selecionar a opção "Expressão regular" na aba de "Pré-processamento" e preencher os parâmetros, por exemplo:

padrão: `".*logged in.*"`, output: `"\0"` - filtra todos os eventos no log de eventos

padrão: `"User (.*)?(?=\s)"`, output: `"\1"` - filtra apenas nomes de usuário do log de eventos

2 Detalhes de pré-processamento

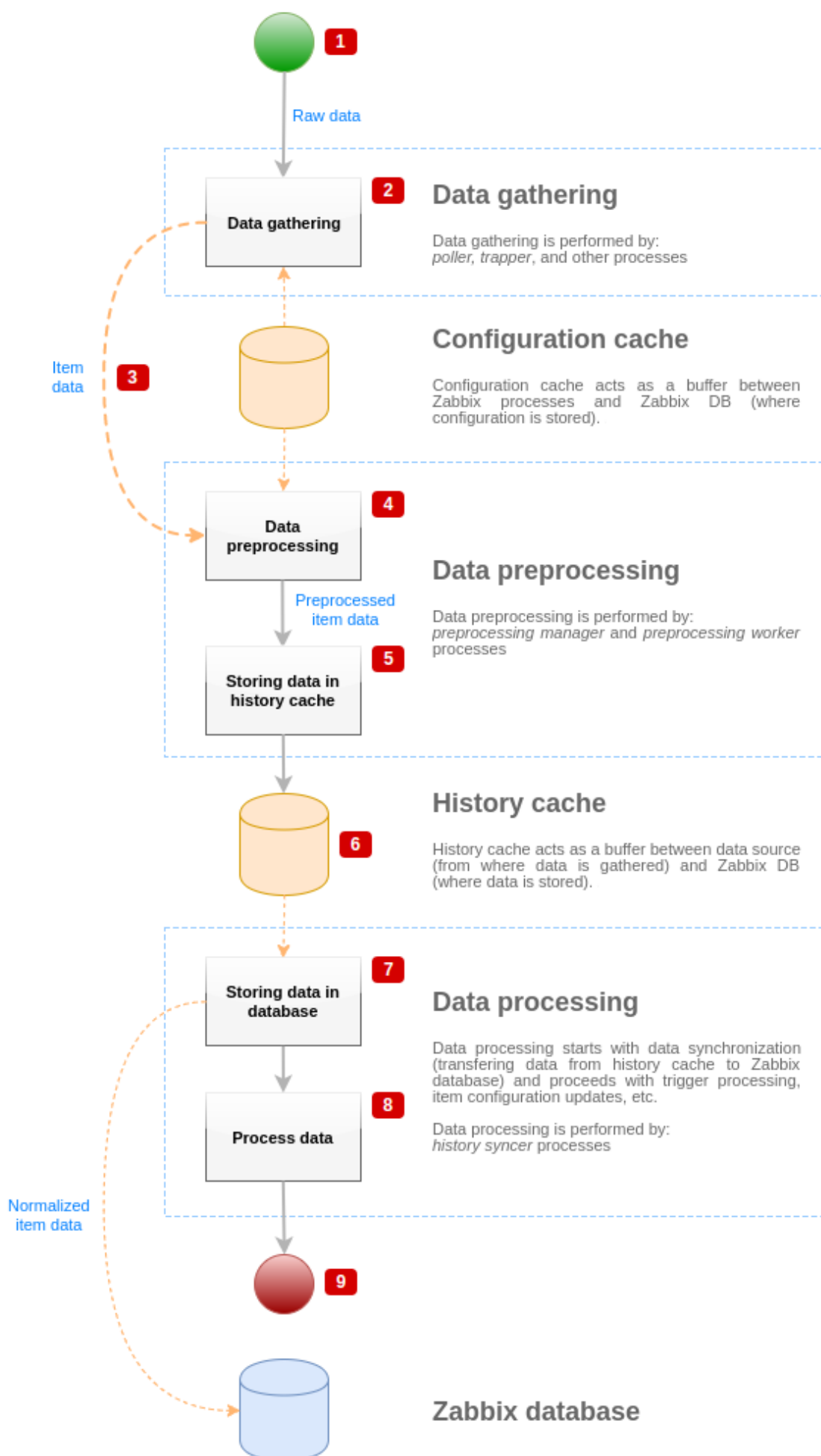
Visão geral

Esta seção fornece detalhes de pré-processamento de valor de item. O pré-processamento de valor de item permite definir e executar **regras de transformação** para os valores de item recebidos.

O pré-processamento é gerenciado por um processo gerenciador de pré-processamento, o qual foi adicionado no Zabbix 3.4, junto com operadores de pré-processamento que executam as etapas de pré-processamento. Todos os valores (com ou sem pré-processamento) de diferentes coletores de dados passam através dos gerenciadores de pré-processamento antes de serem adicionados ao cache histórico. Comunicação IPC baseada em socket é usada entre os coletores de dados (pollers, trappers, etc.) e o processo de pré-processamento. Ou o Zabbix Server ou o Zabbix Proxy (para itens monitorados pelo proxy) está executando etapas de pré-processamento.

Processamento de valor de item

Para visualizar o fluxo de dados de sua origem para o banco de dados do Zabbix, nós podemos usar o seguinte diagrama simplificado:



O diagrama acima mostra apenas processos, objetos e ações relacionadas ao processamento do valor do item em uma forma **simplificada**. O diagrama não mostra mudanças de direção condicionais, manipulação de erros ou loops. O cache de dados local do gerenciador de pré-processamento também não é mostrado devido não afetar o fluxo de dados diretamente. O objetivo deste diagrama é mostrar os processos envolvidos no processamento de valor do item e o modo como eles interagem.

- A coleta de dados começa com dados brutos de uma fonte de dados. Neste ponto, os dados contêm apenas ID, registro de data (timestamp) e valor (também pode ser múltiplos valores)
- Não importa qual o tipo de coletor de dados é usado, a ideia é a mesma para verificações ativas ou passivas, para itens trapper e etc., pois ele apenas altera o formato do dado e o iniciador de comunicação (ou o coletor de dados está aguardando por uma conexão e dados, ou o coletor de dados inicia a comunicação e requisita os dados). O dado bruto é validado, a configuração do item é recuperada do cache de configuração (o dado é enriquecido com os dados de configuração).
- O mecanismo IPC baseado em socket é usado para passar dados dos coletores de dados ao gerenciador de pré-processamento. Neste ponto o coletor de dados continua a coletar dados sem aguardar pela resposta do gerenciador de pré-processamento.
- O pré-processamento é executado. Isto inclui a execução das etapas de pré-processamento e processamento de itens dependentes.

Note:

O item pode alterar seu estado para NÃO SUPORTADO enquanto o pré-processamento é executado se quaisquer etapas do pré-processamento falhar.

- O dado histórico do cache de dados local do gerenciador de pré-processamento está sendo liberado para dentro do cache histórico.
- Neste ponto o fluxo de dados para até a próxima sincronização do cache histórico (quando o processo sincronizador de histórico executa sincronização de dados).
- O processo de sincronização inicia com normalização de dados armazenando dado no banco de dados do Zabbix. A normalização de dados executa conversões para o tipo de item desejado (tipo definido na configuração do item), incluindo a truncagem de dado textual baseado nos tamanhos pré-definidos permitidos para tais tipos (HISTORY_STR_VALUE_LEN para string, HISTORY_TEXT_VALUE_LEN para texto e HISTORY_LOG_VALUE_LEN para valores de log). O dado está sendo enviado para o banco de dados do Zabbix após a normalização ser efetuada.

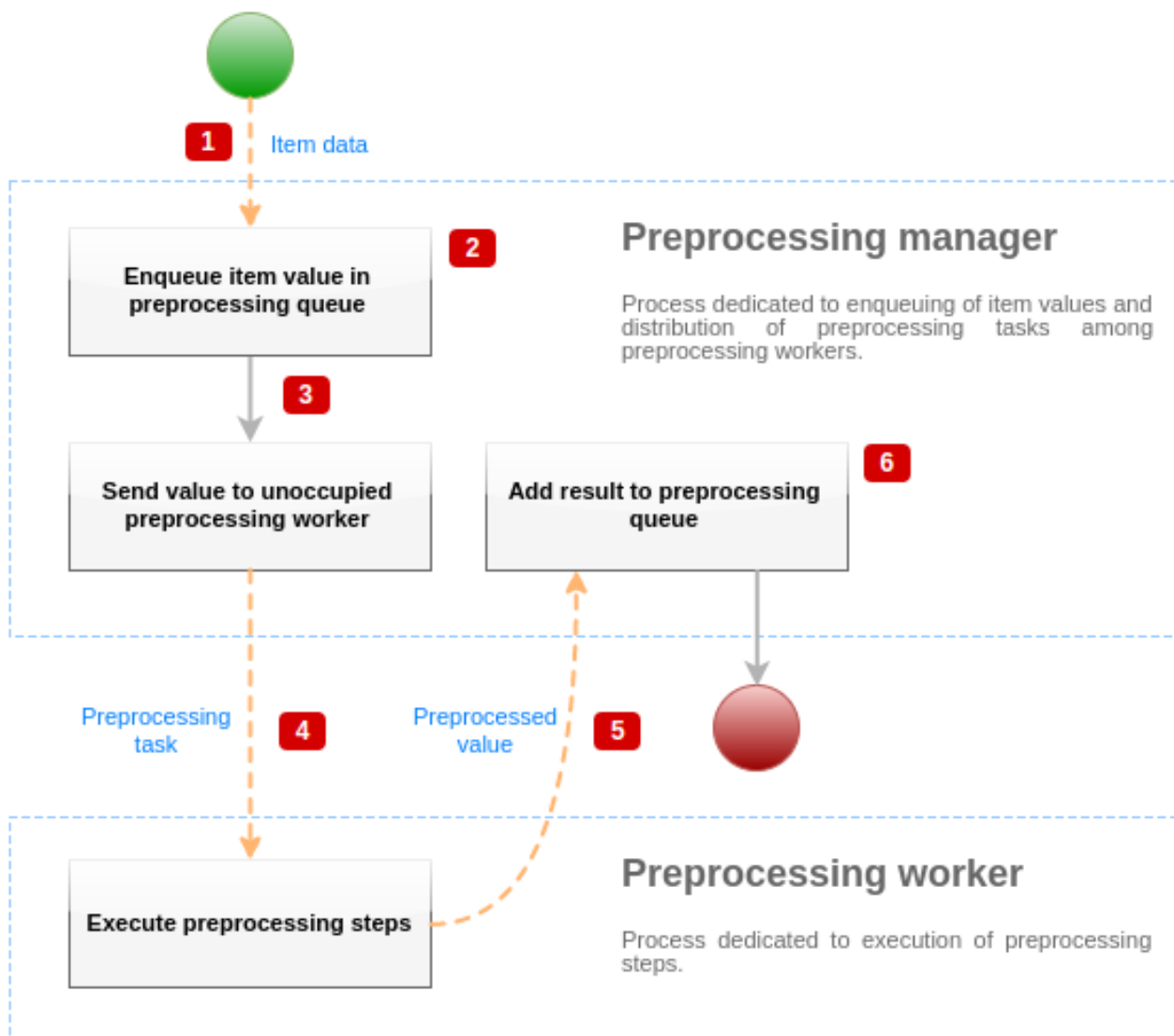
Note:

O item pode alterar seu estado para NÃO SUPORTADO se a normalização de dados falhar (por exemplo, quando um valor textual não pode ser convertido para número).

- O dado coletado está sendo processado - gatilhos são verificados, a configuração do item é atualizada se o item se torna NÃO SUPORTADO, etc.
- Isto é considerado o fim do fluxo de dados do ponto de vista do processamento do valor do item.

Pré-processamento de valor do item

Para visualizar o processo de pré-processamento de dados, nós podemos usar o seguinte diagrama simplificado:



O diagrama acima mostra somente processos, objetos e ações principais relacionadas ao pré-processamento do valor do item em uma forma **simplificada**. O diagrama não mostra alterações de direção condicionais, manipulação de erro ou loops. Apenas um operador de pré-processamento é mostrado neste diagrama (múltiplos operadores de pré-processamento podem ser usados em cenários da vida real), apenas um valor de item está sendo processado e nós assumimos que este item requer a execução de ao menos uma etapa de pré-processamento. O objetivo deste diagrama é mostrar a ideia por detrás da diretiva de pré-processamento do valor do item.

- O dado e valor do item é passado para o gerenciador de pré-processamento usando o mecanismo IPC baseado em socket.
- O item é colocado na fila de pré-processamento.

Note:

O item pode ser colocado no fim ou no início da fila de pré-processamento. Os itens internos do Zabbix são sempre colocados no início da fila de pré-processamento, enquanto outros tipos de itens são enfileirados no fim.

- Neste ponto o fluxo de dados para até que haja ao menos um operador de pré-processamento desocupado (que não esteja executando nenhuma tarefa).
- Quando um operador de pré-processamento está disponível, a tarefa de pré-processamento é enviada para ele.
- Após o pré-processamento ser executado (ambas execuções com falha ou bem-sucedida de etapas de pré-processamento), o valor pré-processado é passado de volta para o gerenciador de pré-processamento.
- O gerenciador de pré-processamento converte o resultado para o formato desejado (definido pelo tipo de valor do item) e coloca o resultado na fila de pré-processamento. Se houver itens dependentes para o item atual, então os itens dependentes são adicionados à fila de pré-processamento também. Itens dependentes são enfileirados na fila de pré-processamento logo após o item principal, mas apenas para itens principais com valor definido e não no estado NÃO SUPORTADO.

Diretiva de processamento de valor

O processamento de valor de item é executado em múltiplas etapas (ou fases) por múltiplos processos. Isto pode causar:

- Item dependente pode receber valores, enquanto O valor principal não. Isto pode ser alcançado pelo uso do seguinte caso de uso:
 - Item principal tem tipo de valor UINT, (item trapper pode ser usado), item dependente tem tipo de valor TEXT.
 - Nenhuma etapa de pré-processamento é requerida para ambos os itens principal e dependente.
 - Valor textual (como, "abc") deve ser passado para o item principal.
 - Como não há etapas de pré-processamento para executar, o gerenciador de pré-processamento verifica se o item principal não está no estado NÃO SUPORTADO e se o valor não está definido (ambos são verdadeiro) e enfileira o item dependente com o mesmo valor do item principal (como não há etapas de pré-processamento).
 - Quando ambos os itens principal e dependente alcançam a fase de sincronização de histórico, o item principal se torna NÃO SUPORTADO, por causa de erro de conversão de valor (dado textual não pode ser convertido para inteiro - unsigned).

Como resultado, o item dependente recebe um valor, enquanto o item principal altera seu estado para NÃO SUPORTADO.

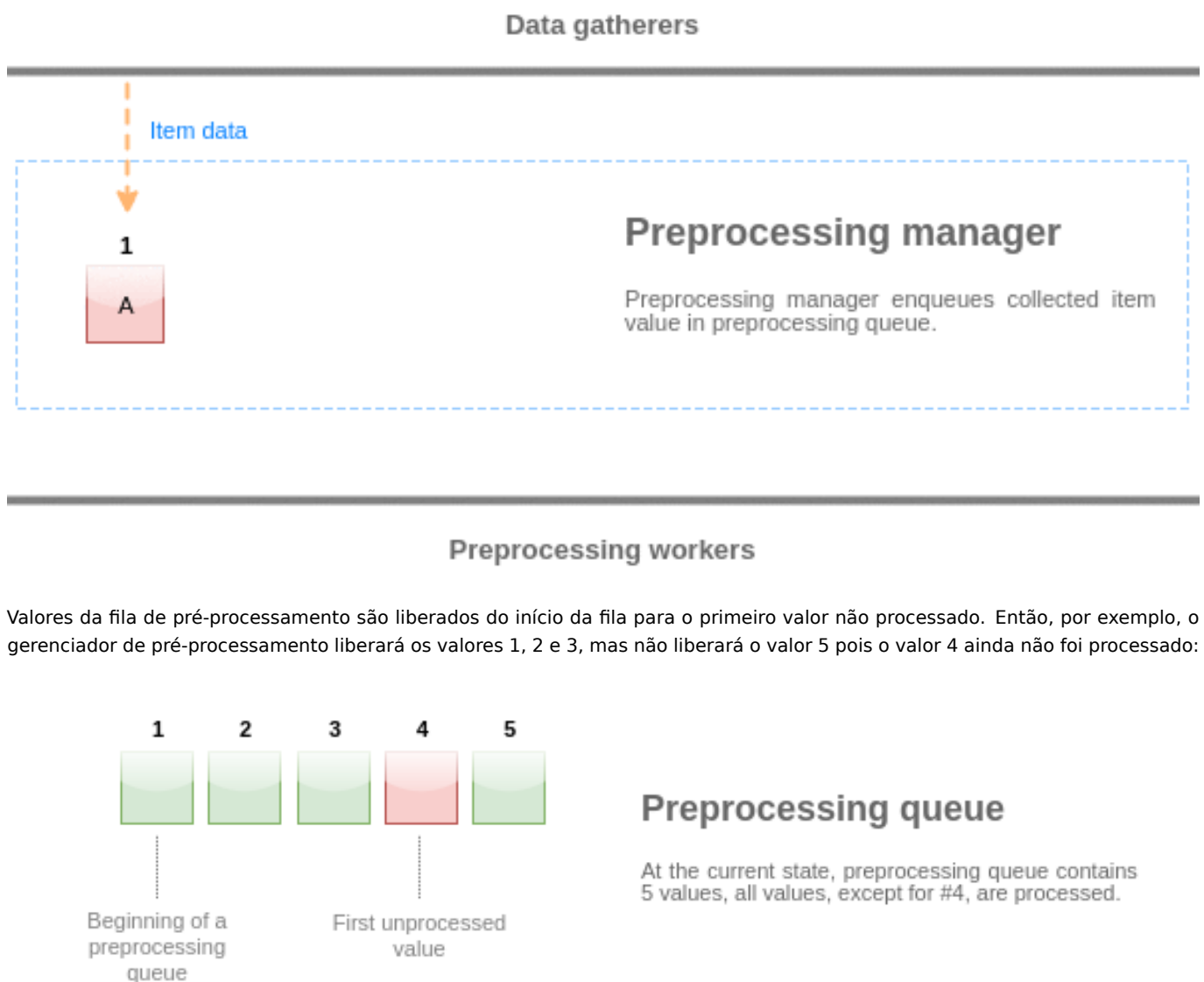
- O item dependente recebe um valor que não está presente no histórico do item principal. O caso de uso é muito similar ao anterior, exceto pelo tipo de item principal. Por exemplo, se o tipo CHAR é usado para o item principal, então o valor do item principal será truncado na fase de sincronização do histórico, enquanto os itens dependentes receberão seu valor do valor inicial (não truncado) do item principal.

Fila de pré-processamento

A fila de pré-processamento é uma estrutura de dados FIFO (primeiro a entrar é o primeiro a sair) que armazena valores preservando a ordem na qual os valores são recebidos pelo gerenciador de pré-processamento. Há múltiplas exceções à lógica FIFO:

- Itens internos são enfileirados no início da fila
- Itens dependentes são sempre enfileirados após os item principal

Para visualizar a lógica da fila de pré-processamento, nós podemos usar o seguinte diagrama:



Apenas dois valores serão deixados na fila (4 e 5) após a liberação, os valores são adicionados ao cache de dados local do gerenciador de pré-processamento e então os valores são transferidos do cache local para dentro do cache histórico. O gerenciador de pré-processamento pode liberar valores do cache de dados local no modo de item único ou no modo volume (usado para itens dependentes e valores recebidos em volume (bulk)).

Operadores de pré-processamento

O arquivo de configuração do Zabbix Server permite aos usuários definir o número de processos de operador de pré-processamento. O parâmetro de configuração `StartPreprocessors` deve ser usado para definir o número de instâncias paralelas (pre-forked) de operadores de pré-processamento. O número ideal de operadores de pré-processamento pode ser determinado por vários fatores, incluindo o número de itens "pré-processáveis" (itens que requerem a execução de quaisquer etapas de pré-processamento), número de processos de coleta de dados, média do número de etapas de pré-processamento para item, etc.

Mas assumindo que não há operações de pré-processamento pesadas como análise de grandes conjuntos XML / JSON, o número de operadores de pré-processamento pode corresponder ao número de coletores de dados. Deste modo, haverá na maioria das vezes (exceto para os casos quando os dados do coletor vierem em volume) ao menos um operador de pré-processamento desocupado para os dados coletados.

Warning:

O excesso de processos de coleta de dados (pollers, pollers inalcançáveis, pollers ODBC, pollers HTTP, pollers Java, operadores de ping, trappers, pollers de proxy) junto com gerenciador IPMI, trapper SNMP e operadores de pré-processamento podem exaurir o limite de descritores de arquivo por processo para o gerenciador de pré-processamento. Isto causará a parada do Zabbix server (usualmente logo após o início, mas algumas vezes isto pode levar um pouco mais de tempo). O arquivo de configuração deve ser revisado ou o limite deve ser elevado para evitar esta situação.

3 Funcionalidade JSONPath

Visão geral

Esta seção fornece detalhes da funcionalidade JSONPath suportado em etapas de pré-processamento de valores de item.

O JSONPath consiste de segmentos separados com pontos. Um segmento pode ser ou uma simples palavra como nome de valor JSON, * ou um caso mais complexo envolto em colchetes []. O ponto de separação antes do segmento em colchete é opcional e pode ser omitido. Por exemplo:

Path (caminho)	Descrição
<code>\$.object.name</code>	Retorna o conteúdo de object.name.
<code>\$.object['name']</code>	Retorna o conteúdo de object.name.
<code>\$.object.['name']</code>	Retorna o conteúdo de object.name.
<code>\$["object"]['name']</code>	Retorna o conteúdo de object.name.
<code>\$.['object'].["name"]</code>	Retorna o conteúdo de object.name.
<code>\$.object.history.length()</code>	Retorna o número de elementos do array object.history.
<code>\$[?(@.name == 'Object')].price.first()</code>	Retorna o campo preço (price) do primeiro objeto com nome 'Object'.
<code>\$[?(@.name == 'Object')].history.first().length()</code>	Retorna o número de elementos do array 'history' do primeiro objeto com nome 'Object'.
<code>\$[?(@.price > 10)].length()</code>	Retorna o número de objetos com preço (price) sendo maior que 10.

Veja também: [Escapando caracteres especiais de valores de macro LLD no JSONPath](#).

Segmentos suportados

Segmento	Descrição
<code><name></code>	Corresponde propriedade do objeto por nome.
<code>*</code>	Corresponde todas as propriedades do objeto.
<code>['<name>']</code>	Corresponde propriedade de objeto por nome.
<code>['<name>', '<name>', ...]</code>	Corresponde propriedade do objeto por qualquer dos nomes listados.
<code>[<index>]</code>	Corresponde elemento do array pelo índice.
<code>[<number>, <number>, ...]</code>	Corresponde elemento do array por qualquer índice listado.
<code>[*]</code>	Corresponde todas as propriedades do objeto ou elementos do array.

Segmento	Descrição
[<start>:<end>]	Corresponde elementos do array pelo intervalo definido: <start> - o primeiro índice para corresponder (inclusive). Se não especificado faz a correspondência de todos os elementos do array desde o início. Se negativo especifica o início de deslocamento a partir do fim do array. <end> - o último índice para corresponder (exclusive). Se não especificado faz a correspondência de todos os elementos do array até o fim. Se negativo especifica o início de deslocamento a partir do fim do array.
[?(<expression>)]	Corresponde elementos de objetos/array pela aplicação de expressão de filtro.

Para encontrar um segmento correspondente ignorando sua ancestralidade (segmento destacado) deve ser prefixado com '..', por exemplo \$. .name ou \$. . ['name'] retorna valores de todas as propriedades 'name'.

Nomes de elemento correspondidos podem ser extraídos pela adição de um sufixo ~ ao JSONPath. Ele retorna o nome do objeto correspondido ou um índice no formato string do item de array correspondente. O formato de saída segue as mesmas regras das outras consultas JSONPath - resultados de caminho definido são retornados 'como são' e resultados de caminho não definido são retornados em array. No entanto não há muita lógica em extrair o nome de um elemento correspondente a um caminho definido - ele já é conhecido.

Expressão de filtro

A expressão de filtro é uma expressão aritmética em notação infix.

Operandos suportados:

Operando	Descrição	Exemplo
"<text>"	Constante de texto.	'value: \'1\'"
'<text>'		"value: '1'"
<number>	Constant numérica suportando notação científica.	123
<jsonpath starting with \$>	Valor referenciado pelo JSONPath do nó raiz do documento de entrada; apenas caminhos (paths) definidos são suportados.	\$.object.name
<jsonpath starting with @>	Valor referenciado pelo JSONPath do objeto/elemento atual; apenas caminhos (paths) definidos são suportados.	@.name

Operadores suportados:

Operador	Tipo	Descrição	Resultado
-	binário	Subtração.	Número.
+	binário	Adição.	Número.
/	binário	Divisão.	Número.
*	binário	Multiplicação.	Número.
==	binário	É igual a.	Booleano. (1 or 0).
!=	binário	Não é igual a.	Booleano. (1 or 0).
	binário	É menor que.	Booleano. (1 or 0).
<=	binário	É menor ou igual a.	Booleano. (1 or 0).
>	binário	É maior que.	Booleano. (1 or 0).
>=	binário	É maior ou igual a.	Booleano. (1 or 0).
=~	binário	Corresponde expressão regular.	Booleano. (1 or 0).
!	unário	Booleano. não (not).	Booleano. (1 or 0).
\\ \\	binário	Booleano. ou (or).	Booleano. (1 or 0).
&&	binário	Booleano. e (and).	Booleano. (1 or 0).

Funções

Funções podem ser usadas no fim do JSONPath. Múltiplas funções podem ser amarradas se a função precedente retorna valor que é aceito pela função seguinte.

Funções suportadas:

Função	Descrição	Entrada	Saída
avg	Valor médio de números no array de entrada.	Array de números.	Número.
min	Valor mínimo de números no array de entrada.	Array de números.	Número.
max	Valor máximo de números no array de entrada.	Array de números.	Número.
sum	Soma de números no array de entrada.	Array de números.	Número.
length	Número de elementos no array de entrada.	Array.	Número.
first	O primeiro elemento do array.	Array.	Uma construct JSON (objeto, array, valor) dependendo do conteúdo do array de entrada.

Valores numéricos quotados são aceitos pelas funções agregadas JSONPath. Isto significa que os valores são convertidos do tipo string para numérico se agregação é necessária.

Uma entrada incompatível causará geração de erro pela função.

Valor de saída

Os JSONPaths podem ser divididos em caminhos definidos e indefinidos. Um caminho definido pode retornar apenas null ou uma simples correspondência. Um caminho indefinido pode retornar múltiplas correspondências, basicamente JSONPaths com múltiplas listas de nome/índice, parte de array ou segmentos de expressão destacados. No entanto, quando uma função é usada o JSONPath se torna definido, pois as funções sempre entregam valores únicos.

Um caminho definido retorna o objeto/array/valor que ele está referenciando, enquanto um caminho indefinido retorna um array dos objetos/arrays/valores correspondidos.

Espaço em branco

O espaço em branco (espaço, caracteres de tabulação) pode ser usado livremente em segmentos e expressões com notação em colchetes, por exemplo, `$['a'][0][?($.b == 'c')][: -1].first()`.

Strings

As strings devem ser quotadas (envoltas) com aspas simples ' ou duplas ". Dentro das strings, as aspas simples ou duplas (dependendo de quais são usadas para quotá-la) e contrabarras \ são escapadas com o caracter contrabarra \.

Exemplos

Dado de entrada

```
{
  "books": [
    {
      "category": "reference",
      "author": "Nigel Rees",
      "title": "Sayings of the Century",
      "price": 8.95,
      "id": 1
    },
    {
      "category": "fiction",
      "author": "Evelyn Waugh",
      "title": "Sword of Honour",
      "price": 12.99,
      "id": 2
    },
    {
      "category": "fiction",
      "author": "Herman Melville",
      "title": "Moby Dick",

```

```

    "isbn": "0-553-21311-3",
    "price": 8.99,
    "id": 3
  },
  {
    "category": "fiction",
    "author": "J. R. R. Tolkien",
    "title": "The Lord of the Rings",
    "isbn": "0-395-19395-8",
    "price": 22.99,
    "id": 4
  }
],
"services": {
  "delivery": {
    "servicegroup": 1000,
    "description": "Next day delivery in local town",
    "active": true,
    "price": 5
  },
  "bookbinding": {
    "servicegroup": 1001,
    "description": "Printing and assembling book in A5 format",
    "active": true,
    "price": 154.99
  },
  "restoration": {
    "servicegroup": 1002,
    "description": "Various restoration methods",
    "active": false,
    "methods": [
      {
        "description": "Checmical cleaning",
        "price": 46
      },
      {
        "description": "Pressing pages damaged by moisture",
        "price": 24.5
      },
      {
        "description": "Rebinding torn book",
        "price": 99.49
      }
    ]
  }
]
},
"filters": {
  "price": 10,
  "category": "fiction",
  "no filters": "no \"filters\""
},
"closed message": "Store is closed",
"tags": [
  "a",
  "b",
  "c",
  "d",
  "e"
]
}

```

JSONPath	Tipo	Resultado	Comentários
\$.filters.price	definido	10	
\$.filters.category	definido	fiction	
\$.filters['no filters']	definido	no "filters" (sem filtros)	
\$.filters	definido	{ "price": 10, "category": "fiction", "no filters": "no \"filters\""} }	
\$.books[1].title	definido	Sword of Honour	
\$.books[-1].author	definido	J. R. R. Tolkien	
\$.books.length()	definido	4	
\$.tags[:]	indefinido	["a", "b", "c", "d", "e"]	
\$.tags[2:]	indefinido	["c", "d", "e"]	
\$.tags[:3]	indefinido	["a", "b", "c"]	
\$.tags[1:4]	indefinido	["b", "c", "d"]	
\$.tags[-2:]	indefinido	["d", "e"]	
\$.tags[: -3]	indefinido	["a", "b"]	
\$.tags[: -3].length()	definido	2	
\$.books[0, 2].title	indefinido	["Sayings of the Century", "Moby Dick"]	
\$.books[1]['author', 'title']	indefinido	["Evelyn Waugh", "Sword of Honour"]	
\$.id	indefinido	[1, 2, 3, 4]	
\$.services..price	indefinido	[5, 154.99, 46, 24.5, 99.49]	
\$.books[?(@.id == 4 - 0.4 * 5)].title	indefinido	["Sword of Honour"]	Esta consulta mostra que operações aritméticas podem ser usadas em consultas. É claro que esta consulta pode ser simplificada para \$.books[?(@.id == 2)].title
\$.books[?(@.id == 2 @.id == 4)].title	indefinido	["Sword of Honour", "The Lord of the Rings"]	
\$.books[?!(@.id == 2)].title	indefinido	["Sayings of the Century", "Moby Dick", "The Lord of the Rings"]	
\$.books[?(@.id != 2)].title	indefinido	["Sayings of the Century", "Moby Dick", "The Lord of the Rings"]	
\$.books[?(@.title =~ " of ")].title	indefinido	["Sayings of the Century", "Sword of Honour", "The Lord of the Rings"]	
\$.books[?(@.price > 12.99)].title	indefinido	["The Lord of the Rings"]	
\$.books[?(@.author > "Herman Melville")].title	indefinido	["Sayings of the Century", "The Lord of the Rings"]	
\$.books[?(@.price > \$.filters.price)].title	indefinido	["Sword of Honour", "The Lord of the Rings"]	
\$.books[?(@.category == \$.filters.category)].title	indefinido	["Sword of Honour", "Moby Dick", "The Lord of the Rings"]	

JSONPath	Tipo	Resultado	Comentários
<code>\$..[?(@.id)]</code>	indefinido	[{ "category": "reference", "author": "Nigel Rees", "title": "Sayings of the Century", "price": 8.95, "id": 1 }, { "category": "fiction", "author": "Evelyn Waugh", "title": "Sword of Honour", "price": 12.99, "id": 2 }, { "category": "fiction", "author": "Herman Melville", "title": "Moby Dick", "isbn": "0-553-21311-3", "price": 8.99, "id": 3 }, { "category": "fiction", "author": "J. R. R. Tolkien", "title": "The Lord of the Rings", "isbn": "0-395-19395-8", "price": 22.99, "id": 4 }]]	
<code>\$.services..[?(@.price > 50)].description</code>	indefinido	['Printing and assembling book in A5 format', 'Rebinding torn book']	
<code>\$..id.length()</code>	definido	4	
<code>\$.books[?(@.id == 2)].title.first()</code>	definido	Sword of Honour	
<code>\$..tags.first().length()</code>	definido	5	<code>\$..tags</code> é um caminho indefinido, então ele retorna um array de elementos correspondidos - [["a", "b", "c", "d", "e"]], <code>first()</code> retorna o primeiro elemento - ["a", "b", "c", "d", "e"] e finalmente <code>length()</code> calcula seu comprimento - 5.
<code>\$.books[*].price.min()</code>	definido	8.95	
<code>\$.price.max()</code>	definido	154.99	
<code>\$.books[?(@.category == "fiction")].price.avg()</code>	definido	14.99	
<code>\$.books[?(@.category == \$.filters.xyz)].title</code>	indefinido		Uma consulta sem correspondência retorna NULL para caminhos definido e indefinido.

JSONPath	Tipo	Resultado	Comentários
<code>\$.services[?(@.active=="true")]definition[1000,1001]</code>	Definição		Constantes de texto devem ser usadas em comparações de valor booleano.
<code>\$.services[?(@.active=="false")]definition[1002]</code>	Definição		Constantes de texto devem ser usadas em comparações de valor booleano.
<code>\$.services[?(@.servicegroup=="def002")]~.first().restoration</code>	Definição		

Escapando caracteres especiais de valores de macro LLD em JSONPath

Quando macros de descoberta de baixo-nível (LLD) são usadas em pré-processamento JSONPath e seus valores são resolvidos, as seguintes regras de escape de caracteres especiais são aplicadas:

- apenas caracteres de contrabarra (\) e aspa duplas (") são considerados para escape;
- se o valor resolvido da macro contém estes caracteres, cada um deles é escapado com uma contrabarra;
- se eles já estão escapados com uma contrabarra, isto não é considerado como escape e ambos a contrabarra e os caracteres especiais são escapados mais uma vez.

Por exemplo:

JSONPath	valor da macro LLD	Após substituição
<code>\$.[?(@.value == "{#MACRO}")]</code>	special "value"	<code>\$.[?(@.value == "special \"value\"")]</code>
	c:\temp	<code>\$.[?(@.value == "c:\\temp")]</code>
	a\\b	<code>\$.[?(@.value == "a\\\\b")]</code>

Quando usada na expressão a macro que pode conter caracteres especiais deve ser quotada com aspas duplas:

JSONPath	valor da macro LLD	Após substituição	Resultado
<code>\$.[?(@.value == "{#MACRO}")]</code>	special "value"	<code>\$.[?(@.value == "special \"value\"")]</code>	OK
<code>\$.[?(@.value == {#MACRO})]</code>		<code>\$.[?(@.value == special \"value\")]</code>	Expressão JSONPath ruim

Quando usada no caminho (path) a macro que pode conter caracteres especiais deve ser quotada com colchetes e aspas duplas:

JSONPath	valor da macro LLD	Após substituição	Resultado
<code>\$.["{#MACRO}"].value</code>	c:\temp	<code>\$.["c:\\temp"].value</code>	OK
<code>\$.{#MACRO}.value</code>		<code>\$.c:\\temp.value</code>	Expressão JSONPath ruim

4 Pré-processamento JavaScript

Visão geral

Esta seção fornece detalhes de pré-processamento por JavaScript.

Pré-processamento JavaScript

O pré-processamento JavaScript é feito pela invocação da função JavaScript com um único parâmetro 'valor' e um corpo de função fornecido pelo usuário. O resultado da etapa de pré-processamento é o valor retornado desta função, por exemplo, para executar a conversão de Fahrenheit para Celsius o usuário deve informar:

```
return (valor - 32) * 5 / 9
```

nos parâmetros de pré-processamento JavaScript, que será embutida em uma função JavaScript pelo Server:

```
function (valor)
{
    return (valor - 32) * 5 / 9
}
```

O parâmetro de entrada 'valor' é sempre passado como um string. O valor de retorno é automaticamente forçado para um string através do método ToString() (se ele falhar então o erro é retornado como um valor string), com algumas poucas exceções:

- retornando valor indefinido resultará em um erro
- retornando valor null fará com que o valor de entrada seja descartado, muito semelhante ao pré-processamento 'Descartar valor' em uma ação 'Customizado em caso de falha'.

Erros podem ser retornados pelo lançamento de valores/objetos (normalmente strings ou objetos Erro).

Por exemplo:

```
if (valor == 0)
    throw "Valor de entrada zero"
return 1/valor
```

Cada script tem um tempo limite de execução de 10 segundos (dependendo do script pode ser levado mais tempo que o limite para engatilhar); excedê-lo retornará um erro. Um limite de heap de 64 megabyte é imposto.

O bytecode da etapa de pré-processamento JavaScript é guardado em cache e reutilizado quando a etapa é aplicada na próxima vez. Quaisquer alterações às etapas de pré-processamento do item farão com que o script em cache seja redefinido e recompilado posteriormente.

Falhas em tempo de execução consecutivas (3 em sequência) farão com que o mecanismo seja reiniciado para mitigar a possibilidade de um script quebrar a execução dos próximos scripts (esta ação é registrada em log com DebugLevel 4 e maior).

O pré-processamento JavaScript é implementado com o mecanismo JavaScript Duktape (<https://duktape.org/>).

Veja também: [Objetos JavaScript adicionais e funções globais](#)

Utilização de macros nos scripts

É possível usar macros de usuário no código JavaScript. Se um script contém macros de usuário, estas macros são resolvidas pelo Server/Proxy antes da execução das etapas de pré-processamento específicas. Note que quando testando etapas de pré-processamento no Frontend, os valores de macro não serão puxados e precisam ser informados manualmente.

Note:

O contexto é ignorado quando uma macro é substituída com seu valor. O valor da macro é inserido no código como está, não é possível adicionar escapes adicionais antes de colocar o valor no código JavaScript. Por favor, esteja ciente que isto pode causar erros de JavaScript em alguns casos.

Em um exemplo abaixo, se o valor recebido excede um valor de {\$THRESHOLD} de macro, o valor de threshold (se presente) será retornado em vez disso:

```
var threshold = '{$THRESHOLD}';
return (!isNaN(threshold) && value > threshold) ? threshold : value;
```

Objetos JavaScript adicionais

Visão geral

Esta seção descreve as adições à linguagem JavaScript implementadas com Duktape e funções JavaScript globais suportadas.

Objetos nativos (built-in)

Zabbix

O objeto Zabbix fornece interação com as funcionalidades internas do Zabbix.

Método	Descrição
log(loglevel, mensagem)	Grava a <mensagem> no log do Zabbix usando o nível de log <loglevel> (veja o parâmetro DebugLevel do arquivo de configuração).

Exemplo:

```
Zabbix.log(3, "esta é uma entrada de log gravada com nível de log 'Warning'")
```

Você pode usar os seguintes apelidos (aliases):

Apelido	Apelido para
<code>console.log(object)</code>	<code>Zabbix.log(4, JSON.stringify(object))</code>
<code>console.warn(object)</code>	<code>Zabbix.log(3, JSON.stringify(object))</code>
<code>console.error(object)</code>	<code>Zabbix.log(2, JSON.stringify(object))</code>

Método	Descrição
<code>sleep(delay)</code>	Atrasa a execução do JavaScript por delay milissegundos.

Exemplo (atrasa execução por 15 segundos):

```
Zabbix.sleep(15000)
```

Requisição HTTP `HttpRequest`

Este objeto encapsula um gerenciamento cURL permitindo a execução de requisições HTTP simples. Erros são lançados como exceção.

Attention:

`HttpRequest` é um novo nome para este objeto desde o Zabbix 5.4. Previamente ele costumava se chamar `CurlHttpRequest`. Os nomes de método também foram alterados no Zabbix 5.4. Os objetos/nomes de método antigos são agora obsoletos e seu suporte será descontinuado após o Zabbix 6.0.

Método	Descrição
<code>addHeader(name, value)</code>	Adicione o campo de cabeçalho HTTP. Este campo é usado para todas as seguintes requisições até que limpas com o método <code>clearHeader()</code> .
<code>clearHeader()</code>	Limpa cabeçalho HTTP. Se nenhum campo cabeçalho é definido, <code>HttpRequest</code> definirá <code>Content-Type</code> como <code>application/json</code> se o dado sendo postado está em formato JSON (JSON-formatted); caso contrário <code>text/plain</code> .
<code>connect(url)</code>	Envia requisição HTTP CONNECT para a URL e retorna a resposta.
<code>customRequest(method, url, data)</code>	Permite especificar qualquer método HTTP no primeiro parâmetro. Envia a requisição do método para a URL com a carga de dados opcionais e retorna a resposta.
<code>delete(url, data)</code>	Envia requisição HTTP DELETE para a URL com a carga de dados opcionais e retorna a resposta.
<code>getHeaders()</code>	Retorna objeto dos campos de cabeçalho HTTP recebidos.
<code>get(url, data)</code>	Envia requisição HTTP GET para a URL com a carga de dados opcionais e retorna a resposta.
<code>head(url)</code>	Envia requisição HTTP HEAD para a URL e retorna a resposta.
<code>options(url)</code>	Envia requisição HTTP OPTIONS para a URL e retorna a resposta.
<code>patch(url, data)</code>	Envia requisição HTTP PATCH para a URL com a carga de dados opcionais e retorna a resposta.
<code>put(url, data)</code>	Envia requisição HTTP PUT para a URL com a carga de dados opcionais e retorna a resposta.
<code>post(url, data)</code>	Envia requisição HTTP POST para a URL com a carga de dados opcionais e retorna a resposta.
<code>getStatus()</code>	Retorna o código de estado da última requisição HTTP.
<code>setProxy(proxy)</code>	Define HTTP proxy para valor "proxy". Se este parâmetro estiver vazio então nenhum proxy é usado.
<code>setHttpAuth(bitmask, username, password)</code>	Define métodos de autenticação HTTP habilitados (HTTPAUTH_BASIC, HTTPAUTH_DIGEST, HTTPAUTH_NEGOTIATE, HTTPAUTH_NTLM, HTTPAUTH_NONE) no parâmetro 'bitmask'. A marcação HTTPAUTH_NONE permite desabilitar a autenticação HTTP. Exemplos: <code>request.setHttpAuth(HTTPAUTH_NTLM HTTPAUTH_BASIC, username, password)</code> <code>request.setHttpAuth(HTTPAUTH_NONE)</code>

Método	Descrição
<code>trace(url, data)</code>	Envia requisição HTTP TRACE para a URL com a carga de dados opcionais e retorna a resposta.

Exemplo:

```
try {
  Zabbix.log(4, 'jira webhook script value='+value);

  var result = {
    'tags': {
      'endpoint': 'jira'
    }
  },
  params = JSON.parse(value),
  req = new HttpRequest(),
  fields = {},
  resp;

  req.addHeader('Content-Type: application/json');
  req.addHeader('Authorization: Basic '+params.authentication);

  fields.summary = params.summary;
  fields.description = params.description;
  fields.project = {"key": params.project_key};
  fields.issuetype = {"id": params.issue_id};
  resp = req.post('https://tsupport.zabbix.lan/rest/api/2/issue/',
    JSON.stringify({"fields": fields})
  );

  if (req.getStatus() != 201) {
    throw 'Response code: '+req.getStatus();
  }

  resp = JSON.parse(resp);
  result.tags.issue_id = resp.id;
  result.tags.issue_key = resp.key;
} catch (error) {
  Zabbix.log(4, 'jira issue creation failed json : '+JSON.stringify({"fields": fields}));
  Zabbix.log(4, 'jira issue creation failed : '+error);

  result = {};
}

return JSON.stringify(result);
```

XML

O objeto XML permite o processamento de dados XML no item e pré- processamento de descoberta de baixo-nível e webhooks.

Attention:

De modo a usar o objeto XML, o Server/Proxy deve ser compilado com suporte a libxml2.

Método	Descrição
<code>XML.query(expression, data)</code>	Recupera conteúdo do nó usando XPath. Retorna null se o nó não for encontrado. expression - uma expressão XPath; data - dado XML como uma string.
<code>XML.toJson(data)</code>	Converte dado em formato XML para JSON.
<code>XML.fromJson(object)</code>	Converte dado em formato JSON para XML.

Exemplo:

Entrada (input):

```
<menu>
  <food type = "breakfast">
    <name>Chocolate</name>
    <price>$5.95</price>
    <description></description>
    <calories>650</calories>
  </food>
</menu>
```

Saída (output):

```
{
  "menu": {
    "food": {
      "@type": "breakfast",
      "name": "Chocolate",
      "price": "$5.95",
      "description": null,
      "calories": "650"
    }
  }
}
```

Regras de serialização

A conversão de XML para JSON será processada de acordo com as seguintes regras (para conversões JSON para XML regras reversas são aplicadas):

1. atributos XML serão convertidos para chaves que possuem seus nomes precedidos com '@'.

Exemplo:

Entrada (input):

```
<xml foo="F00">
  <bar>
    <baz>BAZ</baz>
  </bar>
</xml>
```

Saída (output):

```
{
  "xml": {
    "@foo": "F00",
    "bar": {
      "baz": "BAZ"
    }
  }
}
```

2. Elementos com auto fechamento (<foo/>) serão convertidos como tendo valor 'null'.

Exemplo:

Entrada (input):

```
<xml>
  <foo/>
</xml>
```

Saída (output):

```
{
  "xml": {
    "foo": null
  }
}
```

3. Atributos vazios (com valor "") serão convertidos como tendo um valor string vazia (").

Exemplo:

Entrada (input):

```
<xml>
  <foo bar="" />
</xml>
```

Saída (output):

```
{
  "xml": {
    "foo": {
      "@bar": ""
    }
  }
}
```

4. Múltiplos nós filhos com o mesmo nome de elemento serão convertidos para uma chave única que tem um array de valores como seu valor.

Exemplo:

Entrada (input):

```
<xml>
  <foo>BAR</foo>
  <foo>BAZ</foo>
  <foo>QUX</foo>
</xml>
```

Saída (output):

```
{
  "xml": {
    "foo": ["BAR", "BAZ", "QUX"]
  }
}
```

5. Se um elemento texto não tem atributos e filhos, ele será convertido como uma string.

Exemplo:

Entrada (input):

```
<xml>
  <foo>BAZ</foo>
</xml>
```

Saída (output):

```
{
  "xml": {
    "foo": "BAZ"
  }
}
```

6. Se um elemento texto não tem filhos, mas tem atributos: o conteúdo de texto será convertido para um elemento com a chave 'text' e conteúdo como valor; atributos serão convertidos como descritos na serialização regra 1.

Exemplo:

Entrada (input):

```
<xml>
  <foo bar="BAR">
    BAZ
  </foo>
</xml>
```

Saída (output):

```
{
  "xml": {
    "foo": {
      "@bar": "BAR",
      "#text": "BAZ"
    }
  }
}
```

Funções JavaScript globais

Funções JavaScript globais adicionais foram implementadas com Duktape:

- `btoa(string)` - codifica string para uma string base64
- `atob(base64_string)` - decodifica uma string base64

```
try {
  b64 = btoa("utf8 string");
  utf8 = atob(b64);
}
catch (error) {
  return {'error.name' : error.name, 'error.message' : error.message}
}
```

- `md5(string)` - calcula o hash MD5 de uma string
- `sha256(string)` - calcula o SHA256 de uma string

5 Pré-processamento CSV para JSON

Visão geral

Nesta etapa de pré-processamento é possível converter arquivo de dado CSV para formato JSON. É suportado em:

- itens (protótipos de item)
- regras de descoberta de baixo-nível

Configuração

Para configurar uma etapa de pré-processamento CSV para JSON:

- Vá até a aba de Pré-processamento na configuração [item/regra de descoberta](#)
- Clique em Adicionar
- Selecione a opção CSV para JSON

Preprocessing steps	Name	Parameters
1:	CSV to JSON	<input type="text"/> <input type="text"/> <input checked="" type="checkbox"/> With header

Custom on fail: Discard value Set value to Set error to error messa

[Add](#)

O primeiro parâmetro permite definir um delimitador customizado. Note que se a primeira linha da entrada CSV começa com "Sep=" e é seguida por um caractere UTF-8 único esse caractere será usado como o delimitador no caso de o primeiro parâmetro não estar definido. Se o primeiro parâmetro não estiver definido e um delimitar não for recuperado da linha "Sep=", então uma vírgula é usada como separador.

O segundo parâmetro opcional permite definir um símbolo para quotação.

Se a caixa Com linha de cabeçalho estiver marcada, o valor da linha de cabeçalho será interpretada como nomes de coluna (veja [Processamento de cabeçalho](#) para mais informações).

Se a caixa Customizado em caso de falha estiver marcada, o item não se tornará não suportado no caso de uma etapa de pré-processamento falhar. Adicionalmente opções de manipulação de erro podem ser definidas: descarte o valor, defina um valor especificado ou defina uma mensagem de erro especificada.

Processamento de cabeçalho

A linha de cabeçalho do arquivo CSV pode ser processada em duas formas diferentes:

- Se a caixa Com linha de cabeçalho estiver marcada - os valores da linha são interpretados como nomes de coluna. Neste caso os nomes de coluna devem ser únicos e a linha de dado não deve conter mais colunas do que a linha de cabeçalho;
- Se a caixa Com linha de cabeçalho não estiver marcada - a linha de cabeçalho é interpretada como dado. Os nomes de coluna são gerados automaticamente (1,2,3,4...)

Exemplo de arquivo CSV:

```
Nr,Item name,Key,Qty
1,active agent item,agent.hostname,33
"2","passive agent item","agent.version","44"
3,"active,passive agent items",agent.ping,55
```

Note:

Um caracter de quotação dentro de um campo quotado na entrada deve ser escapado precedendo-o com outro caracter de quotação.

Processando linha de cabeçalho

Saída JSON quando uma linha de cabeçalho é esperada:

```
[
  {
    "Nr": "1",
    "Item name": "active agent item",
    "Key": "agent.hostname",
    "Qty": "33"
  },
  {
    "Nr": "2",
    "Item name": "passive agent item",
    "Key": "agent.version",
    "Qty": "44"
  },
  {
    "Nr": "3",
    "Item name": "active,passive agent items",
    "Key": "agent.ping",
    "Qty": "55"
  }
]
```

Sem processamento de linha de cabeçalho

Saída JSON quando uma linha de cabeçalho não é esperada:

```
[
  {
    "1": "Nr",
    "2": "Item name",
    "3": "Key",
    "4": "Qty"
  },
  {
    "1": "1",
    "2": "active agent item",
    "3": "agent.hostname",
    "4": "33"
  },
  {
    "1": "2",
    "2": "passive agent item",
    "3": "agent.version",
    "4": "44"
  }
]
```

```

},
{
  "1": "3",
  "2": "active,passive agent items",
  "3": "agent.ping"
  "4": "55"
}
]

```

4 História e tendências

Visão geral

Histórico e tendências são as duas formas de armazenar dados coletados no Zabbix.

Enquanto o histórico mantém cada valor coletado, as tendências mantêm a média informações de hora em hora e, portanto, consomem menos recursos.

Mantendo o histórico

Você pode definir por quantos dias o histórico será mantido:

- nas propriedades do item **form**
- ao atualizar itens em massa
- quando **configuração up** tarefas de governanta

Quaisquer dados mais antigos serão removidos pela governanta.

O forte conselho geral é manter o histórico pelo menor tempo possível número de dias e assim não sobrecarregar o banco de dados com muitos valores históricos.

Em vez de manter um longo histórico, você pode manter dados mais longos de tendências. Por exemplo, você pode manter o histórico por 14 dias e as tendências por 5 anos.

Você pode ter uma boa ideia de quanto espaço é necessário para o histórico versus dados de tendências consultando o **dimensionamento do banco de dados page**.

Mantendo um histórico mais curto, você ainda poderá revisar dados em gráficos, pois os gráficos usarão valores de tendência para exibir dados.

::: não importante Se o histórico estiver definido como '0', o item será atualizado apenas itens dependentes e estoque. Nenhuma função de gatilho será avaliado porque a avaliação do gatilho é baseada em dados do histórico só. :::

Como alternativa para preservar história considere usar **história export** funcionalidade de módulos carregáveis.

Mantendo as tendências

O Trends é um mecanismo integrado de redução de dados históricos que armazena mínimo, máximo, médio e o número total de valores por cada hora para tipos de dados numéricos.

Você pode definir por quantos dias as tendências serão mantidas:

- nas propriedades do item **form**
- ao atualizar itens em massa
- ao configurar tarefas de governanta

As tendências geralmente podem ser mantidas por muito mais tempo do que a história. Quaisquer dados mais antigos será removido pela governanta.

O servidor Zabbix acumula dados de tendências em tempo de execução no cache de tendências, conforme os dados fluem. O servidor libera tendências de **hora anterior** de cada item no banco de dados (onde o frontend pode encontrá-los) nestas situações:

- o servidor recebe o valor da primeira hora atual do item
- Faltam 5 minutos ou menos da hora atual e ainda não há hora atual valores do item
- servidor parado

Para ver as tendências em um gráfico, você precisa esperar pelo menos até o início do próxima hora (se o item for atualizado com frequência) e no máximo até o final do a próxima hora (se o item for atualizado raramente), que é de 2 horas no máximo.

Quando o servidor libera o cache de tendências e já existem tendências no banco de dados para esta hora (por exemplo, o servidor foi reiniciado meio da hora), o servidor precisa usar instruções de atualização em vez de simples inserções. Portanto, em uma

instalação maior, se for necessário reiniciar, é desejável parar o servidor no final de uma hora e iniciar no início da próxima hora para evitar sobreposição de dados de tendência.

As tabelas de histórico não participam da geração de tendências de forma alguma.

::: não importante Se as tendências forem definidas como '0', o servidor Zabbix não calcular ou armazenar tendências. :::

Note:

As tendências são calculadas e armazenadas com o mesmo tipo de dados como os valores originais. Como resultado, os cálculos do valor médio de valores de tipo de dados sem sinal são arredondados e quanto menor o intervalo de valor menos preciso será o resultado. Por exemplo, se o item tiver valores 0 e 1, o valor médio será 0, não 0,5.

Além disso, reiniciar o servidor pode resultar na perda de precisão de arquivos não assinados cálculos do valor médio do tipo de dados para a hora atual.

5 Parâmetros do usuário

Visão geral

Às vezes você pode querer executar uma verificação de agente que não vem predefinido com Zabbix. É aqui que os parâmetros do usuário vêm para ajudar.

Você pode escrever um comando que recupere os dados que você precisa e inclua-os no parâmetro do usuário na **configuração do agente arquivo** ('UserParameter' parâmetro de configuração).

Um parâmetro de usuário tem a seguinte sintaxe:

UserParameter=<chave>,<comando>

Como você pode ver, um parâmetro de usuário também contém uma chave. A chave será necessário ao configurar um item. Digite uma chave de sua escolha que ser fácil de referenciar (deve ser único dentro de um host).

Reinicie o agente ou use o agente **runtime control** para pegar o novo parâmetro, e. g.:

```
zabbix_agentd -R userparameter_reload
```

Então, ao **configurar um item**, digite a chave para referenciar o comando do parâmetro do usuário que você deseja executar.

Os parâmetros do usuário são comandos executados pelo agente Zabbix. Até 512 KB de os dados podem ser retornados antes das etapas de pré-processamento do item. Observe, no entanto, que o valor do texto que pode ser eventualmente armazenado no banco de dados é limitado para 64 KB no MySQL (veja informações sobre outros bancos de dados no **tabela**).

/bin/sh é usado como um interpretador de linha de comando em operação UNIX sistemas. Os parâmetros do usuário obedecem ao tempo limite de verificação do agente; se o tempo limite for atingido, o processo de parâmetro de usuário bifurcado é encerrado.

Veja também:

- **Passo a passo tutorial** em usando parâmetros do usuário
- **Execução do comando**

Exemplos de parâmetros de usuário simples

Um comando simples:

```
UserParameter=ping,echo 1
```

O agente sempre retornará '1' para um item com a chave 'ping'.

Um exemplo mais complexo:

```
UserParameter=mysql.ping,mysqladmin -uroot ping | grep -c vivo
```

O agente retornará '1', se o servidor MySQL estiver ativo, '0' - caso contrário.

Parâmetros de usuário flexíveis

Parâmetros de usuário flexíveis aceitam parâmetros com a chave. Desta forma um parâmetro de usuário flexível pode ser a base para a criação de vários itens.

Os parâmetros de usuário flexíveis têm a seguinte sintaxe:

```
UserParameter=chave[*],comando
```

Parâmetro	Descrição
Chave	Chave de item exclusivo. O [*] define que esta chave aceita parâmetros entre colchetes.
Comando	Os parâmetros são fornecidos ao configurar o item. Comando a ser executado para avaliar o valor da chave. Somente para parâmetros de usuário flexíveis: Você pode usar referências posicionais \$1...\$9 no comando para se referir ao respectivo parâmetro em a chave do item. O Zabbix analisa os parâmetros entre [] da chave do item e substitui \$1,...,\$9 no comando de acordo. \$0 será substituído pelo comando original (antes da expansão de \$0,...,\$9) a serem executados. As referências posicionais são interpretadas independentemente de estarem entre aspas duplas ("") ou simples ('). Para usar referências posicionais inalteradas, especifique um double cifrão - por exemplo, awk '{print \$\$2}'. Neste caso, \$\$2 na verdade se transformará em \$2 ao executar o comando.

::: não importante As referências posicionais com o sinal \$ são pesquisadas para e substituído pelo agente Zabbix apenas para parâmetros de usuário flexíveis. Para parâmetros de usuário simples, esse processamento de referência é ignorado e, portanto, nenhuma citação do sinal \$ é necessária. :::

::: não importante Certos símbolos não são permitidos nos parâmetros do usuário por padrão. Ver [UnsafeUserParameters](#) documentação para uma lista completa. :::

Exemplo 1

Algo muito simples:

```
UserParameter=ping[*],echo $1
```

Podemos definir um número ilimitado de itens para monitoramento, todos com formato ping[algo].

- ping[0] - sempre retornará '0'
- ping[aaa] - sempre retornará 'aaa'

Exemplo 2

Vamos adicionar mais sentido!

```
UserParameter=mysql.ping[*],mysqladmin -u$1 -p$2 ping | grep -c vivo
```

Este parâmetro pode ser usado para monitorar a disponibilidade do MySQL base de dados. Podemos passar nome de usuário e senha:

```
mysql.ping[zabbix,our_password]
```

Exemplo 3

Quantas linhas correspondem a uma expressão regular em um arquivo?

```
UserParameter=wc[*],grep -c "$2" $1
```

Este parâmetro pode ser usado para calcular o número de linhas em um arquivo.

```
wc[/etc/passwd,root]
wc[/etc/services,zabbix]
```

Resultado do comando

O valor de retorno do comando é a saída padrão junto com erro padrão.

::: não importante Um texto (caractere, log ou tipo de texto de informações) o item não ficará sem suporte em caso de erro padrão saída. :::

Parâmetros do usuário que retornam texto (caractere, log, tipo de texto de informações) pode retornar espaços em branco. Em caso de resultado inválido o item ficará sem suporte.

1 Estendendo agentes Zabbix

Este tutorial fornece instruções passo a passo sobre como estender o funcionalidade do agente Zabbix com o uso de um **user parâmetro**.

Passo 1

Escreva um script ou linha de comando para recuperar o parâmetro necessário.

Por exemplo, podemos escrever o seguinte comando para obter o total número de consultas executadas por um servidor MySQL:

```
status mysqladmin -uroot | corte -f4 -d":" | corte -f1 -d"S"
```

Quando executado, o comando retorna o número total de consultas SQL.

Passo 2

Adicione o comando ao `zabbix_agentd.conf`:

```
UserParameter=mysql.questions, mysqladmin -uroot status | corte -f4 -d":" | corte -f1 -d"S"
```

mysql.questions é um identificador único. Pode ser qualquer chave válida identificador, por exemplo, consultas.

Teste este parâmetro usando o agente Zabbix com o sinalizador `-t` (se estiver executando em root, no entanto, observe que o agente pode ter permissões diferentes quando lançado como um daemon):

```
zabbix_agentd -t mysql.questions
```

Etapa 3

Recarregue os parâmetros do usuário do arquivo de configuração executando:

```
zabbix_agentd -R userparameter_reload
```

Você também pode reiniciar o agente em vez do comando de controle de tempo de execução.

Teste o parâmetro usando o utilitário **zabbix_get**.

Passo 4

Adicione um novo item com `Key=mysql.questions` ao host monitorado. Tipo de o item deve ser Zabbix Agent ou Zabbix Agent (ativo).

Esteja ciente de que o tipo de valores retornados deve ser definido corretamente no Zabbix servidor. Caso contrário, o Zabbix não os aceitará.

6 Módulos carregáveis

1 Visão geral

Módulos carregáveis oferecem uma opção voltada para o desempenho para estender o Zabbix funcionalidade.

Já existem maneiras de estender a funcionalidade do Zabbix por meio de:

- **parâmetros do usuário** (métricas do agente)
- **verificações externas** (monitoramento sem agente)
- `system.run []` Zabbix **agente item**.

Eles funcionam muito bem, mas têm uma grande desvantagem, ou seja, `fork()`. Zabbix tem que bifurcar um novo processo toda vez que lida com uma métrica de usuário, que é não é bom para o desempenho. Normalmente não é grande coisa, mas pode ser um problema sério ao monitorar sistemas embarcados, tendo um grande número de parâmetros monitorados ou scripts pesados com lógica complexa ou tempo de inicialização longo.

O suporte de módulos carregáveis oferece maneiras de estender o agente Zabbix, servidor e proxy sem sacrificar o desempenho.

Um módulo carregável é basicamente uma biblioteca compartilhada usada pelo daemon do Zabbix e carregado na inicialização. A biblioteca deve conter certas funções, então que um processo Zabbix pode detectar que o arquivo é de fato um módulo que pode carregar e trabalhar.

Os módulos carregáveis têm vários benefícios. Ótimo desempenho e capacidade de implementar qualquer lógica são muito importantes, mas talvez o mais vantagem importante é a capacidade de desenvolver, usar e compartilhar Zabbix módulos. Contribui para uma manutenção sem problemas e ajuda a fornecer nova funcionalidade mais fácil e independente da base de código central do Zabbix.

O licenciamento e distribuição de módulos em formato binário são regidos pela GPL licença (os módulos estão se conectando com o Zabbix em tempo de execução e estão usando o Zabbix cabeçalhos; atualmente todo o código Zabbix está licenciado sob licença GPL). A compatibilidade binária não é garantida pelo Zabbix.

A estabilidade da API do módulo é garantida durante um Zabbix LTS (Long Term Suporte) [lançamento](#) ciclo. A estabilidade da API Zabbix não é garantida (tecnicamente é possível chamar funções internas do Zabbix a partir de um módulo, mas há nenhuma garantia de que tais módulos funcionarão).

2 API do módulo

Para que uma biblioteca compartilhada seja tratada como um módulo Zabbix, é deve implementar e exportar várias funções. Atualmente existem seis funções na API do módulo Zabbix, sendo apenas uma obrigatória e os outros cinco são opcionais.

2.1 Interface obrigatória

A única função obrigatória é **zbx_module_api_version()**:

```
int zbx_module_api_version(void);
```

Esta função deve retornar a versão da API implementada por este módulo e para que o módulo seja carregado esta versão deve corresponder ao módulo Versão da API suportada pelo Zabbix. Versão da API do módulo suportada por Zabbix é ZBX_MODULE_API_VERSION. Então esta função deve retornar isso constante. Constante antiga ZBX_MODULE_API_VERSION_ONE usada para isso O objetivo agora está definido para igualar ZBX_MODULE_API_VERSION para preservar compatibilidade de fonte, mas seu uso não é recomendado.

2.2 Interface opcional

As funções opcionais são **zbx_module_init()**, **zbx_module_item_list()**, **zbx_module_item_timeout()**, **zbx_module_history_write_cbs()** e **zbx_module_uninit()**:

```
int zbx_module_init(void);
```

Esta função deve realizar a inicialização necessária para o módulo (caso existam). Se for bem-sucedido, deve retornar ZBX_MODULE_OK. Caso contrário isto deve retornar ZBX_MODULE_FAIL. Neste último caso, o Zabbix não começa.

```
ZBX_METRIC *zbx_module_item_list(void);
```

Esta função deve retornar uma lista de itens suportados pelo módulo. Cada item é definido em uma estrutura ZBX_METRIC, veja a seção abaixo para detalhes. A lista é terminada por uma estrutura ZBX_METRIC com campo "chave" de NULL.

```
void zbx_module_item_timeout(int timeout);
```

Se o módulo exportar **zbx_module_item_list()** então esta função é usado pelo Zabbix para especificar as configurações de tempo limite na configuração do Zabbix arquivo que as verificações de itens implementadas pelo módulo devem obedecer. Aqui, o parâmetro "timeout" está em segundos.

```
ZBX_HISTORY_WRITE_CBS zbx_module_history_write_cbs(void);
```

Esta função deve retornar funções de callback que o servidor Zabbix usará para histórico de exportação de diferentes tipos de dados. As funções de retorno de chamada são fornecidas como campos da estrutura ZBX_HISTORY_WRITE_CBS, os campos podem ser NULL se módulo não está interessado na história de determinado tipo.

```
int zbx_module_uninit(void);
```

Esta função deve realizar a desinicialização necessária (se houver) como liberar recursos alocados, fechar descritores de arquivos, etc.

Todas as funções são chamadas uma vez na inicialização do Zabbix quando o módulo é carregado, com exceção de zbx_module_uninit(), que é chamado uma vez no desligamento do Zabbix quando o módulo é descarregado.

2.3 Definindo itens

Cada item é definido em uma estrutura ZBX_METRIC:

```
estrutura typedef
{
    caractere *chave;
    bandeiras não assinadas;
    int(*função)();
    char *test_param;
}
ZBX_METRIC;
```

Aqui, **key** é a chave do item (por exemplo, "dummy.random"), **flags** é CF_HAVEPARAMS ou 0 (dependendo se o item aceita parâmetros ou não), **função** é uma função C que implementa o item (por exemplo, "zbx_module_dummy_random") e **test_param** é o lista de parâmetros a ser usada quando o agente Zabbix é iniciado com o "-p" sinalizador (por exemplo, "1,1000", pode ser NULL). Uma definição de exemplo pode parecer esta:

```
chaves ZBX_METRIC estáticas[] =
{
    { "dummy.random", CF_HAVEPARAMS, zbx_module_dummy_random, "1.1000" },
    { NULO }
}
```

Cada função que implementa um item deve aceitar dois ponteiros parâmetros, o primeiro do tipo AGENT_REQUEST e o segundo do tipo digite AGENTE_RESULTADO:

```
int zbx_module_dummy_random(AGENT_REQUEST *solicitação, AGENT_RESULT *resultado)
{
    ...

    SET_UI64_RESULT(resultado, de + rand() % (para - de + 1));

    retornar SYSINFO_RET_OK;
}
```

Essas funções devem retornar SYSINFO_RET_OK, se o valor do item foi obtido com sucesso. Caso contrário, eles devem retornar SYSINFO_RET_FAIL. Veja o exemplo de módulo "dummy" abaixo para obter detalhes sobre como obter informações de AGENT_REQUEST e como definir informações em AGENTE_RESULT.

2.4 Fornecendo retornos de chamada de exportação de histórico

::: não importante A exportação de histórico via módulo não é mais suportada pelo proxy Zabbix desde o Zabbix 4.0.0. :::

O módulo pode especificar funções para exportar dados de histórico por tipo: Numérico (float), Numérico (sem sinal), Caractere, Texto e Log:

```
estrutura typedef
{
    void (*history_float_cb)(const ZBX_HISTORY_FLOAT *history, int history_num);
    void (*history_integer_cb)(const ZBX_HISTORY_INTEGER *history, int history_num);
    void (*history_string_cb)(const ZBX_HISTORY_STRING *history, int history_num);
    void (*history_text_cb)(const ZBX_HISTORY_TEXT *history, int history_num);
    void (*history_log_cb)(const ZBX_HISTORY_LOG *history, int history_num);
}
ZBX_HISTORY_WRITE_CBS;
```

Cada um deles deve ter o array "history" de elementos "history_num" como argumentos. Dependendo do tipo de dados do histórico a ser exportado, "histórico" é uma matriz das seguintes estruturas, respectivamente:

```
estrutura typedef
{
    zbx_uint64_t itemid;
    relógio interno;
    int ns;
    valor duplo;
}
ZBX_HISTORY_FLOAT;
```

```
estrutura typedef
{
    zbx_uint64_t itemid;
    relógio interno;
    int ns;
    valor zbx_uint64_t;
}
ZBX_HISTORY_INTEGER;
```

```
estrutura typedef
{
    zbx_uint64_t itemid;
```

```

    relógio interno;
    int ns;
    const char *valor;
}
ZBX_HISTORY_STRING;

estrutura typedef
{
    zbx_uint64_t itemid;
    relógio interno;
    int ns;
    const char *valor;
}
ZBX_HISTORY_TEXT;

estrutura typedef
{
    zbx_uint64_t itemid;
    relógio interno;
    int ns;
    const char *valor;
    const char *fonte;
    int timestamp;
    int logeventid;
    gravidade int;
}
ZBX_HISTORY_LOG;

```

Callbacks serão usados pelos processos de sincronização de histórico do servidor Zabbix no fim do procedimento de sincronização do histórico após os dados serem gravados no banco de dados Zabbix e salvo no cache de valor.

2.5 Construindo módulos

Atualmente, os módulos devem ser construídos dentro da árvore de origem do Zabbix, porque a API do módulo depende de algumas estruturas de dados que são definidas nos cabeçalhos do Zabbix.

O cabeçalho mais importante para módulos carregáveis é **include/module.h**, que define essas estruturas de dados. Outros cabeçalhos de sistema necessários que ajudar **include/module.h** a funcionar corretamente são **stdlib.h** e **stdint.h**.

Com essas informações em mente, está tudo pronto para que o módulo seja construído. O módulo deve incluir **stdlib.h**, **stdint.h** e **module.h**, e o script de compilação deve garantir que esses arquivos sejam no caminho de inclusão. Veja o exemplo de módulo "fictício" abaixo para obter detalhes.

Outro cabeçalho útil é **include/log.h**, que define **zabbix_log()** função, que pode ser usada para registro e depuração propósitos.

3 Parâmetros de configuração

Agent, servidor e proxy Zabbix suportam dois **parâmetros** para lidar com módulos:

- LoadModulePath - caminho completo para a localização dos módulos carregáveis
- LoadModule - módulo(s) para carregar na inicialização. Os módulos devem ser localizado em um diretório especificado por LoadModulePath ou o caminho deve preceder o nome do módulo. Se o caminho anterior for absoluto (começa com '/') então LoadModulePath é ignorado. É permitido incluir vários parâmetros LoadModule.

Por exemplo, para estender o agente Zabbix, podemos adicionar o seguinte parâmetros:

```

LoadModulePath=/usr/local/lib/zabbix/agent/
LoadModule=mariadb.so
LoadModule=apache.so
LoadModule=kernel.so
LoadModule=/usr/local/lib/zabbix/dummy.so

```

Após a inicialização do agent, ele carregará o mariadb.so, apache.so e kernel.so módulos do diretório /usr/local/lib/zabbix/agent enquanto dummy.so será carregado de /usr/local/lib/zabbix. Ele falhará se um módulo for ausente, em caso de permissões ruins ou se uma biblioteca compartilhada não for um Módulo Zabbix.

4 Configuração de front-end

Módulos carregáveis são suportados pelo agente Zabbix, servidor e proxy. Portanto, o tipo de item no frontend do Zabbix depende de onde o módulo está carregado. Se o módulo estiver carregado no agente, o tipo de item deve ser "agente Zabbix" ou "agente Zabbix (ativo)". Se o módulo estiver carregado no servidor ou proxy, então o tipo de item deve ser "Simple Verifica".

A exportação de histórico através dos módulos Zabbix não precisa de nenhuma configuração. Se o módulo for carregado com sucesso pelo servidor e fornece a função **zbx_module_history_write_cbs()** que retorna pelo menos uma função de retorno de chamada não NULL, a exportação do histórico será habilitado automaticamente.

5 Módulo fictício

O Zabbix inclui um módulo de exemplo escrito em linguagem C. O módulo é localizado em `src/modules/dummy`:

```
alex@alex:~trunk/src/modules/dummy$ ls -l
-rw-rw-r-- 1 alex alex 9019 24 de abril 17:54 dummy.c
-rw-rw-r-- 1 alex alex 67 24 de abril 17:54 Makefile
-rw-rw-r-- 1 alex alex 245 24 de abril 17:54 README
```

O módulo está bem documentado, pode ser usado como modelo para o seu próprio módulos.

Depois que `./configure` foi executado na raiz da árvore de origem do Zabbix como descrito acima, basta executar **make** para compilar **dummy.so**.

```
/*
** Zabbix
** Copyright (C) 2001-2020 Zabbix SIA
**
** This program is free software; you can redistribute it and/or modify
** it under the terms of the GNU General Public License as published by
** the Free Software Foundation; either version 2 of the License, or
** (at your option) any later version.
**
** This program is distributed in the hope that it will be useful,
** but WITHOUT ANY WARRANTY; without even the implied warranty of
** MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
** GNU General Public License for more details.
**
** You should have received a copy of the GNU General Public License
** along with this program; if not, write to the Free Software
** Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.
**/

####include <stdlib.h>
####include <string.h>
####include <time.h>
####include <stdint.h>

####include "module.h"

/* the variable keeps timeout setting for item processing */
static int item_timeout = 0;

/* module SHOULD define internal functions as static and use a naming pattern different from Zabbix intern
/* symbols (zbx_*) and loadable module API functions (zbx_module_*) to avoid conflicts
static int dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result);
static int dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result);
static int dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result);

static ZBX_METRIC keys[] =
/* KEY FLAG FUNCTION TEST PARAMETERS */
{
    {"dummy.ping", 0, dummy_ping, NULL},
    {"dummy.echo", CF_HAVEPARAMS, dummy_echo, "a message"},
    {"dummy.random", CF_HAVEPARAMS, dummy_random, "1,1000"},
    {NULL}
};
```

```

/*****
 *
 * Function: zbx_module_api_version
 *
 * Purpose: returns version number of the module interface
 *
 * Return value: ZBX_MODULE_API_VERSION - version of module.h module is
 *             compiled with, in order to load module successfully Zabbix
 *             MUST be compiled with the same version of this header file
 *
 *****/
int zbx_module_api_version(void)
{
    return ZBX_MODULE_API_VERSION;
}

/*****
 *
 * Function: zbx_module_item_timeout
 *
 * Purpose: set timeout value for processing of items
 *
 * Parameters: timeout - timeout in seconds, 0 - no timeout set
 *
 *****/
void zbx_module_item_timeout(int timeout)
{
    item_timeout = timeout;
}

/*****
 *
 * Function: zbx_module_item_list
 *
 * Purpose: returns list of item keys supported by the module
 *
 * Return value: list of item keys
 *
 *****/
ZBX_METRIC *zbx_module_item_list(void)
{
    return keys;
}

static int dummy_ping(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    SET_UI64_RESULT(result, 1);

    return SYSINFO_RET_OK;
}

static int dummy_echo(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char *param;

    if (1 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters."));
        return SYSINFO_RET_FAIL;
    }
}

```

```

    param = get_rparam(request, 0);

    SET_STR_RESULT(result, strdup(param));

    return SYSINFO_RET_OK;
}

/*****
 *
 * Function: dummy_random
 *
 * Purpose: a main entry point for processing of an item
 *
 * Parameters: request - structure that contains item key and parameters
 *              request->key - item key without parameters
 *              request->nparam - number of parameters
 *              request->params[N-1] - pointers to item key parameters
 *              request->types[N-1] - item key parameters types:
 *                  REQUEST_PARAMETER_TYPE_UNDEFINED (key parameter is empty)
 *                  REQUEST_PARAMETER_TYPE_ARRAY (array)
 *                  REQUEST_PARAMETER_TYPE_STRING (quoted or unquoted string)
 *
 *              result - structure that will contain result
 *
 * Return value: SYSINFO_RET_FAIL - function failed, item will be marked
 *               as not supported by zabbix
 *               SYSINFO_RET_OK - success
 *
 * Comment: get_rparam(request, N-1) can be used to get a pointer to the Nth
 *           parameter starting from 0 (first parameter). Make sure it exists
 *           by checking value of request->nparam.
 *           In the same manner get_rparam_type(request, N-1) can be used to
 *           get a parameter type.
 *****/
static int dummy_random(AGENT_REQUEST *request, AGENT_RESULT *result)
{
    char    *param1, *param2;
    int from, to;

    if (2 != request->nparam)
    {
        /* set optional error message */
        SET_MSG_RESULT(result, strdup("Invalid number of parameters.));
        return SYSINFO_RET_FAIL;
    }

    param1 = get_rparam(request, 0);
    param2 = get_rparam(request, 1);

    /* there is no strict validation of parameters and types for simplicity sake */
    from = atoi(param1);
    to = atoi(param2);

    if (from > to)
    {
        SET_MSG_RESULT(result, strdup("Invalid range specified.));
        return SYSINFO_RET_FAIL;
    }

    SET_UI64_RESULT(result, from + rand() % (to - from + 1));

```

```

    return SYSINFO_RET_OK;
}

/*****
 *
 * Function: zbx_module_init
 *
 * Purpose: the function is called on agent startup
 *          It should be used to call any initialization routines
 *
 * Return value: ZBX_MODULE_OK - success
 *              ZBX_MODULE_FAIL - module initialization failed
 *
 * Comment: the module won't be loaded in case of ZBX_MODULE_FAIL
 *
 *****/
int zbx_module_init(void)
{
    /* initialization for dummy.random */
    srand(time(NULL));

    return ZBX_MODULE_OK;
}

/*****
 *
 * Function: zbx_module_uninit
 *
 * Purpose: the function is called on agent shutdown
 *          It should be used to cleanup used resources if there are any
 *
 * Return value: ZBX_MODULE_OK - success
 *              ZBX_MODULE_FAIL - function failed
 *
 *****/
int zbx_module_uninit(void)
{
    return ZBX_MODULE_OK;
}

/*****
 *
 * Functions: dummy_history_float_cb
 *            dummy_history_integer_cb
 *            dummy_history_string_cb
 *            dummy_history_text_cb
 *            dummy_history_log_cb
 *
 * Purpose: callback functions for storing historical data of types float,
 *          integer, string, text and log respectively in external storage
 *
 * Parameters: history      - array of historical data
 *            history_num - number of elements in history array
 *
 *****/
static void dummy_history_float_cb(const ZBX_HISTORY_FLOAT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

```

```

    }
}

static void dummy_history_integer_cb(const ZBX_HISTORY_INTEGER *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_string_cb(const ZBX_HISTORY_STRING *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_text_cb(const ZBX_HISTORY_TEXT *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

static void dummy_history_log_cb(const ZBX_HISTORY_LOG *history, int history_num)
{
    int i;

    for (i = 0; i < history_num; i++)
    {
        /* do something with history[i].itemid, history[i].clock, history[i].ns, history[i].value, ... */
    }
}

/*****
 *
 * Function: zbx_module_history_write_cbs
 *
 * Purpose: returns a set of module functions Zabbix will call to export
 *          different types of historical data
 *
 * Return value: structure with callback function pointers (can be NULL if
 *              module is not interested in data of certain types)
 *
 *****/
ZBX_HISTORY_WRITE_CBS    zbx_module_history_write_cbs(void)
{
    static ZBX_HISTORY_WRITE_CBS    dummy_callbacks =
    {
        dummy_history_float_cb,
        dummy_history_integer_cb,
        dummy_history_string_cb,
        dummy_history_text_cb,
    }
}

```

```

        dummy_history_log_cb,
    };

    return dummy_callbacks;
}

```

6 Limitações

O suporte de módulos carregáveis é implementado apenas para a plataforma Unix. Isso significa que não funciona para agentes do Windows.

Em alguns casos, um módulo pode precisar ler a configuração relacionada ao módulo parâmetros de `zabbix_agentd.conf`. Não é suportado atualmente. Se você precisa que seu módulo use alguns parâmetros de configuração que você deve provavelmente implementa a análise de um arquivo de configuração específico do módulo.

7 contadores de desempenho do Windows

Visão geral

Você pode monitorar efetivamente os contadores de desempenho do Windows usando o `perf_counter[]` tecla.

Por exemplo:

```
perf_counter["\Processor(0)\Interrupções/s"]
```

ou

```
perf_counter["\Processor(0)\Interrupções/s", 10]
```

Para obter mais informações sobre como usar esta chave ou seu equivalente somente em inglês `perf_counter_en`, consulte [item específico do Windows keys](#).

Para obter uma lista completa de contadores de desempenho disponíveis para monitoramento, você pode executar:

```
typeperf -qx
```

Você também pode usar a descoberta de baixo nível para descobrir vários **object instâncias** dos contadores de desempenho do Windows e automatizar a criação de `perf_counter` itens para vários objetos de instância.

Representação numérica

O Windows mantém representações numéricas (índices) para objetos e nomes de contadores de desempenho. Zabbix suporta essas representações numéricas como parâmetros para as chaves de item `perf_counter`, `perf_counter_en` e em Parâmetros de configuração `PerfCounter`, `PerfCounterEn`.

No entanto, não é recomendado usá-los a menos que você possa garantir seu índices numéricos mapeiam para corrigir strings em hosts específicos. Se você precisar criar itens portáteis que funcionem em diferentes hosts com várias versões localizadas do Windows, você pode usar a chave `perf_counter_en` ou Parâmetro de configuração `PerfCounterEn` que permite usar nomes em inglês independentemente da localidade do sistema.

Para descobrir os equivalentes numéricos, execute **regedit** e encontre `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Perflib\009`.

A entrada do registro contém informações como esta:

```

1
1847
2
Sistema
4
Memória
6
% de tempo do processador
10
Operações de leitura de arquivo/s
12
Operações de Gravação de Arquivo/s
14
Operações de controle de arquivos/s
16
Bytes de leitura de arquivo/s

```

18

Bytes de gravação de arquivo/s

....

Aqui você pode encontrar os números correspondentes para cada parte da string do contador de desempenho, como em '\System\% Processor Time':

Sistema → 2

% de tempo do processador → 6

Então você pode usar esses números para representar o caminho em números:

\2\6

Parâmetros do contador de desempenho

Você pode implantar alguns parâmetros PerfCounter para o monitoramento do Windows contadores de desempenho.

Por exemplo, você pode adicioná-los ao arquivo de configuração do agente Zabbix:

```
PerfCounter=UserPerfCounter1,"\Memory\Page Reads/s",30
ou
PerfCounter=UserPerfCounter2,"4\24",30
```

Com esses parâmetros em vigor, você pode simplesmente usar UserPerfCounter1 ou UserPerfCounter2 como as chaves para criar o respectivos itens.

Lembre-se de reiniciar o agente Zabbix depois de fazer alterações no arquivo de configuração.

8 Atualização em massa

Visão geral

Algumas vezes você precisa alterar um atributo em um grande número de itens de uma só vez. Ao invés de alterar item a item, você pode utilizar o recurso de atualização em massa para fazer esta tarefa.

Usando atualização em massa

Para atualizar alguns itens em massa, faça o seguinte:

- Marque as caixas de seleção dos itens a serem atualizados na lista
- Clique em Atualização em massa abaixo da lista
- Navegue até a guia com os atributos necessários (Item, Tags ou Pré-processando)
- Marque as caixas de seleção dos atributos a serem atualizados
- Insira novos valores para os atributos

Mass update

Item Tags Preprocessing

Private key file ☐ Original

Password ☐ Original

Update interval ☐ Original

History storage period ☒ Do not keep history Storage period 7d

Trend storage period ☐ Original

Status ☐ Original

Log time format ☐ Original

Value mapping ☐ Original

Enable trapping ☐ Original

Mass update

Item Tags Preprocessing

Tags ☒ Add Replace Remove

Name

Value

tag

value

Add

A opção Tags permite:

- Add - adiciona tags especificadas aos itens (tags com o mesmo nome, mas valores diferentes não são considerados 'duplicados' e podem ser adicionado ao mesmo host).
- Substituir - remove as tags especificadas e adiciona tags com novos valores
- Remover - remove tags especificadas dos itens

Macros de usuário, macros {INVENTORY.*}, {HOST.HOST}, {HOST.NAME}, As macros {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} e {HOST.ID} são suportado em tags.

Mass update

Item

Tags

Preprocessing

Preprocessing steps ☒

Name

Parameters



1:

JavaScript



script



2:

JSONPath



\$.path.to.node

Add

Quando terminar, clique em Atualizar.

8 Mapeamento de valores

Visão geral

Para uma representação mais "humana" dos valores recebidos, você pode usar value maps que contêm o mapeamento entre valores numéricos/string e string representações.

Mapeamentos de valor podem ser usados tanto no frontend do Zabbix quanto nas notificações enviados por tipos de mídia.

Por exemplo, um item que tem valor '0' ou '1' pode usar mapeamento de valor para representam os valores em uma forma legível por humanos:

- '0' => 'Não disponível'
- '1' => 'Disponível'

Ou, um mapa de valor relacionado ao backup pode ser:

- 'F' => 'Cheio'
- 'D' => 'Diferencial'
- 'Eu' => 'Incremental'

Em outro exemplo, as faixas de valores para tensão podem ser mapeadas:

- '<=209' => 'Baixo'
- '210-230' => 'OK'
- '>=231' => 'Alto'

Os mapeamentos de valor são definidos no modelo ou no nível do host. Uma vez definidos eles tornam-se disponíveis para todos os itens do respectivo modelo ou host. Lá não há herança de mapa de valor - um item de modelo em um host ainda usa o mapa de valor do modelo; vincular um modelo com mapas de valor ao host não faz com que o host herde os mapas de valores.

Ao **configurar itens** você pode usar um mapa de valores para "humanizar" o forma como um valor de item será exibido. Para fazer isso, você se refere ao nome de um mapa de valores previamente definido no campo Mapeamento de valores.

Note:

O mapeamento de valores pode ser usado com itens com Numeric (sem sinal), Numérico (float) e Caractere tipo de em formação.

Os mapeamentos de valor podem ser exportados/importados com o respectivo modelo ou hospedeiro.

Os mapeamentos de valor podem ser atualizados em massa. Ambos **host** e **template** os formulários de atualização em massa têm um Guia Mapeamento de valores para mapas de valores de atualização em massa.

Configuração

Para definir um mapa de valores:

- Abra um formulário de configuração de host ou modelo
- Vá para a guia Mapeamento de valores

- Clique em Adicionar para adicionar um novo mapa
- Clique no nome de um mapa existente para editá-lo

Value mapping

* Name

VMware status

* Mappings

Type	Value	Mapped to
<div> <div></div> <div>equals</div> <div></div> </div>	0	⇒ gray
<div> <div></div> <div>equals</div> <div></div> </div>	1	⇒ green
<div> <div></div> <div>equals</div> <div></div> </div>	2	⇒ yellow
<div> <div></div> <div>equals</div> <div></div> </div>	3	⇒ red

Add

Update

Parâmetros de um mapa de valores:

Parâmetro	Descrição
Nome	Nome exclusivo de um conjunto de mapeamentos de valor.
Mapeamentos	Regras de mapeamento individual para mapear valores numéricos/string para representações de string. O mapeamento é aplicado de acordo com a ordem das regras de mapeamento. É possível reordenar mapeamentos arrastando. Somente tipos de valores numéricos são suportados para mapeamentos de intervalos (é maior ou igual a, é menor ou igual a, no intervalo tipos de mapeamento).
Tipo	<p>Tipo de mapeamento:</p> <p>igual - valores iguais serão mapeados</p> <p>é maior ou igual a - valores iguais ou maiores serão mapeados</p> <p>é menor ou igual - valores iguais ou menores serão mapeados</p> <p>no intervalo - valores no intervalo serão mapeados; o intervalo é expresso como <number1>-<number2> ou <number>. Vários intervalos são compatíveis (por exemplo, 1-10.101-110.201)</p> <p>regexp - valores correspondentes à expressão regular serão mapeados (expressões regulares globais não são compatíveis)</p> <p>* *padrão* - todos os valores pendentes serão mapeados, exceto aqueles com mapeamentos específicos</p>
Value	<p>Valor de entrada.</p> <p>Dependendo do tipo de mapeamento, também pode conter um intervalo ou expressão regular.</p>
Mapeado para	Representação de string para o valor de entrada.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Quando o mapa de valores é exibido na lista, apenas os três primeiros mapeamentos dele são visíveis, enquanto três pontos

indicam que mais mapeamentos existir.

Template	Linked templates	Tags	Macros 4	Value mapping 1
Name		Value		
VMware status		=0 ⇒ gray		
		=1 ⇒ green		
		=2 ⇒ yellow		
		...		
Add				

Como isso funciona

Por exemplo, um dos itens de agente predefinidos 'ping do agente Zabbix' usa um mapa de valores no nível do modelo chamado 'status de ping do agente Zabbix' para exibir seus valores.

Value mapping

* Name

* Mappings

Type	Value	Mapped to
<div><div></div><div>equals</div></div>	<input type="text" value="1"/>	⇒ <input type="text" value="Up"/>

No item **formulário de configuração** você pode ver uma referência a este mapa de valores no campo Mapeamento de valores:

Value mapping

Select

Portanto, em Monitoring → Últimos dados, o mapeamento é usado para exibir 'Up' (com o valor bruto entre parênteses).

▼ <input type="checkbox"/> Host ▲	Name	Last check	Last value
▼ <u>Zabbix server</u>	Monitoring agent (1 item)		
<input type="checkbox"/>	Zabbix agent ping	02/23/2021 04:27:07 PM	Up (1)

Na seção Latest data, os valores exibidos são reduzidos para 20 símbolos. Se o mapeamento de valor for usado, esse encurtamento não será aplicado ao valor mapeado, mas apenas para o valor bruto separadamente (exibido em parênteses).

Um valor sendo exibido em um formato legível também é mais fácil de entender ao receber notificações.

Sem um mapa de valores predefinido, você só obterá isso:

▼ <input type="checkbox"/> Host ▲	Name	Last check	Last value
▼ <u>Zabbix server</u>	Monitoring agent (1 item)		
<input type="checkbox"/>	Zabbix agent ping	02/23/2021 06:00:07 PM	1

Então, neste caso, você teria que adivinhar o que o '1' significa ou faça uma pesquisa de documentação para descobrir.

10 Fila

Visão geral

A fila apresenta os itens que estão com algum atraso em sua coleta. Ela é uma representação **logical** dos dados. Não existe nenhuma fila "IPC" ou qualquer outro mecanismo de fila no Zabbix.

As estatísticas apresentadas pela fila são um bom indicador de performance do Zabbix Server.

A fila é coletada diretamente do processo do Zabbix Server através do protocolo JSON. A informação só está disponível enquanto o processo do Zabbix Server estiver em execução.

Consultando a fila

Para consultar a fila acesse Administração → Fila. A opção Visão geral pode ser selecionada no canto superior direito da tela.

≡ Queue overview ▾

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	1	11	1	0	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Zabbix internal	0	0	0	0	0	0
Zabbix aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	0	0	0	0	0

As células que estão verdes indicam normalidade naquela faixa de análise. Na imagem acima a fila apresenta um item aguardando por 5 segundos e cinco itens aguardando por 30 segundos. Uma vez que temos estes indicativos, a dúvida normal é: e o que está causando isso? Quais itens estão atrasados?

Para ver o detalhamento dos itens em atraso, selecione Detalhes na caixa de seleção situada no canto superior da tela. Agora você deverá estar vendo uma lista com os itens em atraso.

≡ Queue details ▾

Scheduled check	Delayed by	Host	Name	Proxy
2019-09-02 11:46:40	58s	My host	CPU idle time	Remote proxy
2019-09-02 11:46:41	57s	My host	CPU interrupt time	Remote proxy
2019-09-02 11:46:42	56s	My host	CPU iowait time	Remote proxy
2019-09-02 11:46:43	55s	My host	CPU nice time	Remote proxy
2019-09-02 11:46:44	54s	My host	CPU softirq time	Remote proxy
2019-09-02 11:46:45	53s	My host	CPU steal time	Remote proxy
2019-09-02 11:46:46	52s	My host	CPU system time	Remote proxy

A partir da análise destes dados poderá ser possível localizar e corrigir a causa raiz destes atrasos.

Ter um ou dois itens em atraso talvez não seja motivo para alarmes. Eles poderão estar atualizados em poucos segundos. Entretanto, se você visualizar uma grande quantidade de itens em atraso, isso poderá ser um sério problema para sua monitoração.

ITEMS	5 SECONDS	10 SECONDS	30 SECONDS	1 MINUTE	5 MINUTES	MORE THAN 10 MINUTES
Zabbix agent	0	13	7	0	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Zabbix internal	5	1	9	0	0	0

Item da fila

Existe um item especial de uso interno ao Zabbix que responde pela chave **zabbix[queue,<from>,<to>]** e pode ser utilizado para monitorar a saúde da fila do Zabbix. Ele irá retornar a quantidade de itens em atraso em um determinado tempo. Para maiores detalhes favor consultar o manual de **itens internos**.

11 Cache de valores

Visão geral

Para efetuar os cálculos das expressões de triggers, itens calculados, itens agregados, e macros de forma mais ágil o Zabbix (a partir da versão 2.2) suporta manter um cache de valores no Zabbix Server.

Este cache é em memória e pode ser usado para acessar valores históricos, ao invés de proceder com consultas SQL diretas ao banco de dados. Se o histórico de valores não estiver presente no cache, os valores ausentes serão solicitados do BD e o cache será atualizado.

Para habilitar a funcionalidade de cache de valores você deverá definir o parâmetro opcional **ValueCacheSize** no [arquivo de configuração do Zabbix Server](#).

Dois itens internos são suportados para monitorar o cache de valores: **zabbix[vcache,buffer,<mode>]** e **zabbix[vcache,cache,<parameter>]**. Veja mais detalhes em [itens internos](#).

```
#####12 Execute agora {#manual-config-items-check_now}
```

Visão geral

A verificação de um novo valor de item no Zabbix é um processo cíclico que é com base nos intervalos de atualização configurados. Enquanto para muitos itens a atualização intervalos são bastante curtos, existem outros (incluindo regras de descoberta) para os quais os intervalos de atualização são bastante longos, portanto, em situações da vida real, pode ser necessário verificar um novo valor mais rápido - para captar alterações em recursos detectáveis, por exemplo. Para acomodar tal necessidade, é possível reprogramar um verifique e recupere um novo valor imediatamente.

Essa funcionalidade é compatível apenas com verificações **passivas**. O seguintes tipos de itens são suportados:

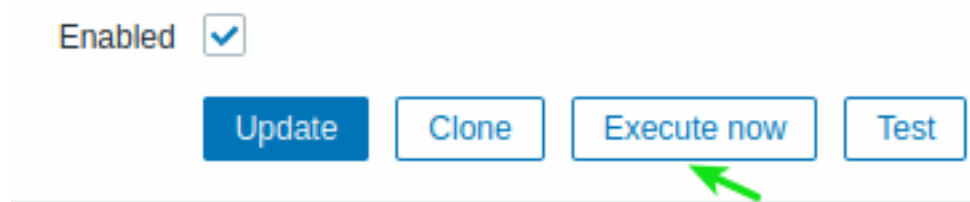
- Agente Zabbix (passivo)
- Agente SNMPv1/v2/v3
- Agente IPMI
- Verificação simples
- Zabbix interno
- Verificação externa
- Monitor de banco de dados
- Agente JMX
- Agente SSH
- Telnet
- Calculado
- Agente HTTP
- Roteiro

... não importante A verificação deve estar presente na configuração cache para ser executado; Para mais informações, veja [CacheUpdateFrequency](#). Antes de executando a verificação, o cache de configuração **não** é atualizado, portanto alterações muito recentes na configuração de regra de item/descoberta não serão pegou. Portanto, também não é possível verificar um novo valor para um item/regra que está sendo criado ou que acabou de ser criado; usar a opção Test enquanto configura um item para isso.
...

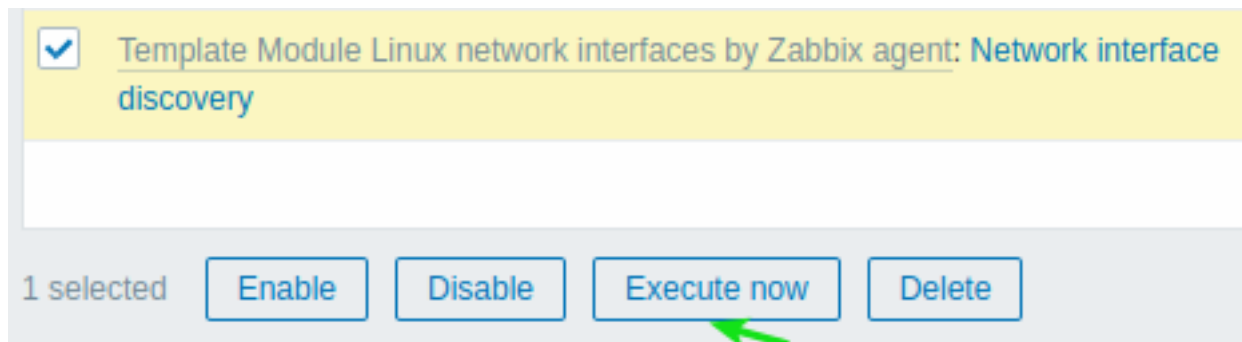
Configuração

Para executar uma verificação passiva imediatamente:

- clique em Executar agora em um item existente (ou regra de descoberta) formulário de configuração:



- clique em Execute now para itens/regras selecionados na lista de itens/regras de descoberta:



Neste último caso vários itens/regras podem ser selecionados e “executados agora” de uma vez.

13 Restringindo verificações de agentes

Visão geral

É possível restringir as verificações do lado do agente criando um item lista negra, uma lista branca ou uma combinação de lista branca/lista negra.

Para fazer isso, use uma combinação de dois agentes **configuração** parâmetros:

- AllowKey=<pattern> - quais verificações são permitidas; <padrão> é especificado usando uma expressão curinga (*)
- DenyKey=<pattern> - quais verificações são negadas; <padrão> é especificado usando uma expressão curinga (*)

Observe que:

- Todos os itens `system.run[*]` (comandos remotos, scripts) são desabilitados por padrão, mesmo quando nenhuma chave de negação é especificada;
- Desde o Zabbix 5.0.2 o parâmetro do agente EnableRemoteCommands é:

- * obsoleto pelo agente Zabbix
- * não suportado pelo agente Zabbix2

Portanto, para permitir todos os comandos remotos, especifique um parâmetro AllowKey=system.run[*]. Para permitir apenas alguns comandos remotos, crie uma lista branca de comandos específicos do system.run[]. Para não permitir comandos remotos específicos, adicione parâmetros DenyKey com system.run[] comandos antes do parâmetro AllowKey=system.run[*].

Regras importantes

- Uma lista branca sem uma regra de negação só é permitida para system.run[*] Itens. Para todos os outros itens, os parâmetros AllowKey não são permitidos sem um parâmetro DenyKey; neste caso o agente Zabbix **não irá start** apenas com parâmetros AllowKey.
- A ordem importa. Os parâmetros especificados são verificados um a um de acordo com sua ordem de aparência no arquivo de configuração:
 - Assim que uma chave de item corresponder a uma regra de permissão/negação, o item será permitido ou negado; e a verificação de regras pára. Então, se um item corresponde a uma regra de permissão e uma regra de negação, o resultado será depende de qual regra vem primeiro.
 - A ordem afeta também o parâmetro EnableRemoteCommands (se usado).
- Números ilimitados de parâmetros AllowKey/DenyKey são suportados.
- As regras AllowKey, DenyKey não afetam HostnameItem, Parâmetros de configuração HostMetadataItem, HostInterfaceItem.
- O padrão de chave é uma expressão curinga em que o curinga (*) caractere corresponde a qualquer número de caracteres em determinada posição. Pode ser usado tanto no nome da chave quanto nos parâmetros.
- Se uma chave de item específica não for permitida na configuração do agente, o item será relatado como não suportado (nenhuma dica é dada quanto ao razão);
- O agente Zabbix com a opção de linha de comando --print (-p) não será exibido chaves que não são permitidas pela configuração;
- O agente Zabbix com a opção de linha de comando --test (-t) retornará “Chave de item não compatível.” status para chaves que não são permitidas por configuração;
- Comandos remotos negados não serão registrados no log do agente (se LogRemoteCommands=1).

Casos de uso

Negar verificação específica

- Lista negra de uma verificação específica com o parâmetro DenyKey. Chaves correspondentes será desautorizado. Todas as chaves não correspondentes serão permitidas, exceto system.run[] itens.

Por exemplo:

```
# Negar acesso seguro a dados
DenyKey=vfs.file.contents[/etc/passwd,*]
```

::: não importante Uma lista negra pode não ser uma boa escolha, porque um nova versão do Zabbix pode ter novas chaves que não são explicitamente restritas pela configuração existente. Isso pode causar uma segurança falha. :::

Negar comando específico, permitir outros

- Lista negra de um comando específico com o parâmetro DenyKey. Colocar todos na lista de permissões outros comandos, com o parâmetro AllowKey.

```
# Desabilita comando específico
DenyKey=system.run[ls -l /]
```

```
# Permitir outros scripts
AllowKey=system.run[*]
```

Permitir verificação específica, negar outras

- Verificações específicas da lista de permissões com parâmetros AllowKey, negue outras com DenyKey=*

Por exemplo:

```
# Permitir a leitura de logs:
AllowKey=vfs.file.*[/var/log/*]
```

```
# Permitir verificações de hora local
AllowKey=system.localtime[*]
```

```
# Negar todas as outras chaves
DenyKey=*
```

Exemplos de padrões

Padrão	Descrição	Correspondências	Sem correspondência
*	Corresponde a todas as chaves possíveis com ou sem parâmetros.	Qualquer	Nenhum
vfs.file.contents	Corresponde a <code>vfs.file.contents</code> sem parâmetros.	vfs.file.contents	vfs.file.contents[/etc/passwd]
vfs.file.contents[]	Corresponde <code>vfs.file.contents</code> com parâmetros vazios.	vfs.file.contents[]	vfs.file.contents
vfs.file.contents[*]	Corresponde <code>vfs.file.contents</code> com quaisquer parâmetros; não corresponderá a <code>vfs.file.contents</code> sem colchetes.	vfs.file.contents[] vfs.file.contents[/path/to/file]	vfs.file.contents
vfs.file.contents[/etc/passwd]	Corresponde <code>vfs.file.contents</code> com os primeiros parâmetros correspondentes a <code>/etc/passwd</code> e todos os outros parâmetros com qualquer valor (também vazio).	vfs .file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]	vfs.file.contents[/etc/passwd] vfs.file.contents[/var/log/zabbix_server.log] vfs.file.contents[]
vfs.file.contents[*passwd*]	Corresponde <code>vfs.file.contents</code> com o primeiro parâmetro correspondente a <code>*passwd*</code> e nenhum outro parâmetro.	vfs.file.contents[/etc/passwd]	vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]
vfs.file.contents[*passwd*,*]	Corresponde a <code>vfs.file.contents</code> apenas com o primeiro parâmetro correspondente a <code>*passwd*</code> e todos os parâmetros seguintes com qualquer valor (também vazio).	vfs.file.contents[/etc/passwd,] vfs.file.contents[/etc/passwd,utf8]	vfs.file.contents[/etc/passwd] vfs.file.contents[/tmp/test] vfs.file.contents[]
vfs.file.contents[/var/log/zabbix_server.log]	Corresponde <code>vfs.file.contents</code> com o primeiro parâmetro correspondente a <code>/var/log/zabbix_server.log</code> , terceiro parâmetro correspondente a <code>'abc'</code> e qualquer segundo parâmetro (também vazio).	vfs.file.contents[/var/log/zabbix_server.log,] vfs.file.contents[/var/log/zabbix_server.log,utf8,abc]	vfs.file.contents[/var/log/zabbix_server.log,] vfs.file.contents[]

Padrão	Descrição	Correspondências	Sem correspondência
<code>vfs.file.contents[/etc/passwd,utf8]</code>	Corresponde a <code>vfs.file.contents</code> com o primeiro parâmetro correspondente a <code>/etc/passwd</code> , segundo parâmetro correspondente a <code>'utf8'</code> e nenhum outro argumento.	<code>vfs.file.contents[/etc/passwd,utf8]</code>	<code>vfs.file.contents[/etc/passwd,]</code> <code>vfs.file.contents[/etc/passwd,utf16]</code>
<code>vfs.file.*</code>	Corresponde a qualquer chave que comece com <code>vfs.file.</code> sem nenhum parâmetro.	<code>vfs.file.contents</code> <code>vfs.file.size</code>	<code>vfs.file.contents[]
vfs.file.size[/var/log/zabbix_server.log</code>
<code>vfs.file.*[*]</code>	Corresponde a qualquer chave que comece com <code>vfs.file.</code> com quaisquer parâmetros.	<code>vfs.file.size.bytes[]</code> <code>vfs.file.size[/var/log/zabbix_server.log,utf8]</code>	<code>vfs.file.size.bytes</code>
<code>vfs.*.contents</code>	Corresponde a qualquer chave começando com <code>vfs.</code> e terminando com <code>.contents</code> sem nenhum parâmetro.	<code>vfs.mount.point.file.contents</code> <code>vfs..contents</code>	<code>vfs. conteúdo</code>

`system.run` e `AllowKey`

Um script hipotético como `'myscript.sh'` pode ser executado em um host via Agente Zabbix de várias maneiras:

1. Como chave de item em uma verificação passiva ou ativa, por exemplo:

- `system.run[myscript.sh]`
- `system.run[myscript.sh,wait]`
- `system.run[myscript.sh.nowait]`

Aqui o usuário pode adicionar `"wait"`, `"nowait"` ou omitir o 2º argumento para usar seu valor padrão em `system.run[]`.

2. Como um script global (iniciado pelo usuário no frontend ou API).

Um usuário configura este script em Administration → Scripts, define `"Execute on: Zabbix agent"` e coloca `"myscript.sh"` no script Campo de entrada `"Comandos"`. Quando invocado do frontend ou API o Zabbix servidor envia ao agente:

- `system.run[myscript.sh,wait]` - até Zabbix 5.0.4
- `system.run[myscript.sh]` - desde 5.0.5

Aqui o usuário não controla os parâmetros `"wait"/"nowait"`.

3. Como um comando remoto de uma ação. O servidor Zabbix envia para agente:

- `system.run[myscript.sh,nowait]`

Aqui, novamente, o usuário não controla os parâmetros `"wait"/"nowait"`.

O que isso significa é se definirmos `AllowKey` como:

```
AllowKey=system.run[myscript.sh]
```

então

- `system.run[myscript.sh]` - será permitido
- `system.run[myscript.sh,wait]`, `system.run[myscript.sh,nowait]` não será permitido - o script não será executado se invocado como passo de ação

Para permitir todas as variantes descritas, você pode adicionar:

```
AllowKey=system.run[myscript.sh,*]
```

```
DenyKey=system.run[*]
```

aos parâmetros agente/agente2.

14 Plugins

Visão geral

Os plugins fornecem uma opção para estender as capacidades de monitoramento do Zabbix. Os plugins são escritos na linguagem de programação Go e são suportados apenas pelo agente Zabbix 2.

Os plugins oferecem uma alternativa aos módulos **loadable modules** (escritos em C), e outros métodos para estender a funcionalidade do Zabbix, como parâmetros do usuário **user parameters** (métricas do agente), **external checks** (monitoramento sem agente), e o item do agente do Zabbix `system.run[]` **agent item**.

As seguintes características são específicas do agente Zabbix 2 e de seus plugins:

- suporte a intervalos programados e flexíveis para verificações tanto passivas quanto ativas;;
- gerenciamento da fila de tarefas com relação ao cronograma e à concorrência de tarefas;
- timeouts no nível do plugin;
- verificação de compatibilidade do agente Zabbix 2 e seu plugin na inicialização..

Desde o Zabbix 6.0.0, os plugins não precisam ser integrados diretamente ao agente 2 e podem ser adicionados como plugins carregáveis, tornando assim o processo de criação de plugins adicionais para coletar novas métricas de monitoramento mais fácil.

Esta página lista os plugins nativos e carregáveis do Zabbix e descreve os princípios de configuração do plugin do ponto de vista do usuário. Para instruções sobre como escrever seus próprios plugins, consulte [Plugin development guidelines](#). Para obter mais informações sobre o processo de comunicação entre o agente Zabbix 2 e um plugin carregável, bem como o processo de coleta de métricas, consulte [Connection diagram](#).

Configurando plugins

Esta seção fornece princípios de configuração comuns e melhores práticas para plugins

Todos os parâmetros são configurados usando o parâmetro `Plugins.* parameter`, que pode fazer parte do Zabbix agent 2 **configuration file** ou do próprio plugin **configuration file**. Se um plugin usar um arquivo de configuração separado, o caminho para este arquivo deve ser especificado no parâmetro `Include` do arquivo de configuração do agente Zabbix 2.

Um parâmetro típico do plugin tem a seguinte estrutura: `Plugins.<PluginName>.<Parameter>=<Value>`

Além disso, existem dois grupos específicos de parâmetros:

- `Plugins.<PluginName>.Default.<Parameter>=<Value>` usado para definir **default parameter values**.
- `Plugins.<PluginName>.<SessionName>.<Parameter>=<Value>` usado para definir conjuntos separados de parâmetros para diferentes alvos de monitoramento através de **named sessions**.

Todos os nomes de parâmetros devem seguir os seguintes requisitos:

- é recomendado que os nomes de seus plugins sejam capitalizados;
- o parâmetro deve ser capitalizado;
- caracteres especiais não são permitidos;
- aninhamento não está limitado por um nível máximo;
- o número de parâmetros não é limitado.

Sessões nomeadas

As sessões nomeadas representam um nível adicional de parâmetros de plugin e podem ser usadas para especificar conjuntos separados de parâmetros de autenticação para cada uma das instâncias sendo monitoradas. Cada parâmetro de sessão nomeada deve ter a seguinte estrutura:

`Plugins.<PluginName>.Sessions.<SessionName>.<Parameter>=<Value>`

Um nome de sessão pode ser usado como um parâmetro de item chave `connString` em vez de especificar um URI, nome de usuário e/ou senha separadamente.

Nos parâmetros de chave de item, o primeiro parâmetro pode ser um `connString` ou uma URI. Se o primeiro parâmetro da chave não corresponder a nenhum nome de sessão, ele será tratado como um URI. Note que embutir credenciais em um URI não é suportado, use parâmetros de sessão nomeada em seu lugar.

Essa lista **named session parameters** disponível depende do plugin.

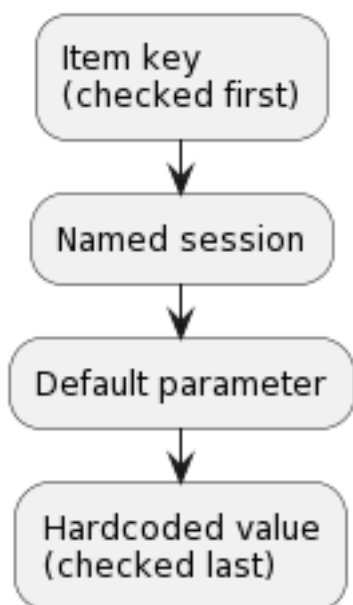
No Zabbix antes da versão 6.0.17, ao fornecer um `connString` (nome da sessão) nos parâmetros da chave do item, os parâmetros do nome de usuário e senha devem estar vazios. Os valores serão retirados dos parâmetros da sessão. Se um parâmetro de autenticação não for especificado para a sessão nomeada, um valor padrão fixo será usado.

Desde o Zabbix 6.0.17, é possível substituir os parâmetros da sessão especificando novos valores nos parâmetros da chave do item (consulte **example**).

Desde o Zabbix 6.0.18, se um parâmetro não estiver definido para a sessão nomeada, Zabbix agent 2 usará o valor definido no parâmetro padrão de login **default plugin parameter**.

Prioridade de parâmetro

Desde a versão 6.0.18, os plugins do Zabbix agent 2 procuram por valores de parâmetros relacionados à conexão na seguinte ordem:



1. O primeiro parâmetro da chave do item é comparado com os nomes das sessões. Se nenhuma correspondência for encontrada, ele será tratado como um valor real; neste caso, a etapa 3 será ignorada. Se uma correspondência for encontrada, o valor do parâmetro (geralmente, um URI) deve ser definido na sessão nomeada.
2. Outros parâmetros serão retirados da chave do item se estiverem definidos.
3. Se um parâmetro da chave do item (por exemplo, senha) estiver vazio, o plugin procurará pelo parâmetro de sessão nomeada correspondente.
4. Se o parâmetro da sessão também não estiver especificado, o valor definido no parâmetro padrão correspondente **default parameter** será usado.
5. Se tudo falhar, o plugin usará o valor padrão codificado.

Exemplo 1

Monitoramento de duas instâncias “MySQL1” e “MySQL2”.

Parâmetros de configuração:

```

Plugins.Mysql.Sessions.MySQL1.Uri=tcp://127.0.0.1:3306
Plugins.Mysql.Sessions.MySQL1.User=mysql1_user
Plugins.Mysql.Sessions.MySQL1.Password=unique_password
Plugins.Mysql.Sessions.MySQL2.Uri=tcp://192.0.2.0:3306
Plugins.Mysql.Sessions.MySQL2.User=mysql2_user
Plugins.Mysql.Sessions.MySQL2.Password=different_password
  
```

Como resultado desta configuração, cada nome da sessão pode ser usado como um connString em uma **item key**, por exemplo, `mysql.ping[MySQL1]` ou `mysql.ping[MySQL2]`.

Exemplo 2

Fornecendo alguns dos parâmetros da chave do item (suportado desde o Zabbix 6.0.17).

Parâmetros de configuração:

```

Plugins.PostgreSQL.Sessions.Session1.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Sessions.Session1.User=old_username
Plugins.PostgreSQL.Sessions.Session1.Password=session_password
  
```

Item key: `pgsql.ping[session1,new_username,,postgres]`

Como resultado desta configuração, o agente se conectará ao PostgreSQL usando os seguintes parâmetros:

- URI do parâmetro da sessão: 192.0.2.234:5432
- Nome de usuário da chave do item: new_username
- Senha do parâmetro da sessão (já que é omitida na chave do item): session_password
- Nome do banco de dados da chave do item: postgres

Exemplo 3

Coletando uma métrica usando parâmetros de configuração padrão:

Parâmetros de configuração:

```
Plugins.PostgreSQL.Default.Uri=tcp://192.0.2.234:5432
Plugins.PostgreSQL.Default.User=zabbix
Plugins.PostgreSQL.Default.Password=password
```

Item key: pgsql.ping[, , , postgres]

Como resultado desta configuração, o agente se conectará ao PostgreSQL usando os seguintes parâmetros:

- URI padrão: 192.0.2.234:5432
- Nome de usuário padrão: zabbix
- Senha padrão: password
- Nome do banco de dados da chave do item: postgres

Valores padrão

Desde o Zabbix 6.0.18, você pode definir valores padrão para os parâmetros relacionados à conexão (URI, nome de usuário, senha, etc.) no arquivo de configuração no formato:

Plugins.<PluginName>.Default.<Parameter>=<Value>

Por exemplo, Plugins.Mysql.Default.Username=zabbix, Plugins.MongoDB.Default.Uri=tcp://127.0.0.1:27017, etc.

Se um valor para tal parâmetro não for fornecido em uma chave de item ou nos parâmetros da **named session**, plugin usará o valor padrão. Se um parâmetro padrão também não estiver definido, serão usados valores padrão codificados.

Se uma chave de item não tiver nenhum parâmetro, o agente Zabbix 2 tentará coletar a métrica usando os valores definidos na seção de parâmetros padrão.

Conexões

Alguns plugins suportam a coleta de métricas de várias instâncias simultaneamente. Tanto instâncias locais quanto remotas podem ser monitoradas. Conexões TCP e Unix-socket são suportadas.

É recomendável configurar os plugins para manter as conexões com as instâncias em um estado aberto. Os benefícios incluem redução de congestionamento de rede, latência e uso de CPU e memória devido ao menor número de conexões. A biblioteca do cliente cuida disso.

Note:

O período de tempo pelo qual conexões não utilizadas devem permanecer abertas pode ser determinado pelo parâmetro Plugins.<PluginName>.KeepAlive.

Exemplo: Plugins.Memcached.KeepAlive

Plugins

Todas as métricas suportada pelo Zabbix agent 2 são coletadas pelos plugins.

Integrados

Os seguintes plugins para o agente Zabbix 2 estão disponíveis e prontos para uso. Clique no nome do plugin para acessar o repositório do plugin com informações adicionais.

Nome do plugin	Descrição	Supported item keys	Comentarios
Agente	Métricas do agente Zabbix sendo utilizado.	agent.hostname, agent.ping, agent.version	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
Ceph	Monitoramento Ceph.	ceph.df.details, ceph.osd.stats, ceph.osd.discovery, ceph.osd.dump, ceph.ping, ceph.pool.discovery, ceph.status	
CPU	Monitoramento da CPU do sistema (número de CPUs/núcleos de CPU, CPUs descobertas, percentual de utilização).	system.cpu.discovery, system.cpu.num, system.cpu.util	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .

Nome do plugin	Descrição	Supported item keys	Comentarios
Docker	Monitoramento dos containers Docker.	docker.container_info, docker.container_stats, docker.containers, docker.containers.discovery, docker.data_usage, docker.images, docker.images.discovery, docker.info, docker.ping	Consulte também: Configuration parameters
Arquivo	Coleta de métricas de arquivo.	vfs.file.cksum, vfs.file.contents, vfs.file.exists, vfs.file.md5sum, vfs.file.regexp, vfs.file.regmatch, vfs.file.size, vfs.file.time	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
Kernel	Monitoramento de kernel.	kernel.maxfiles, kernel.maxproc	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
Log	Monitoramento de arquivos de log.	log, log.count, logrt, logrt.count	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys . Consulte também: Parâmetros de configuração do Plugin (Unix/Windows)
Memcached	Monitoramento do servidor Memcached.	memcached.ping, memcached.stats	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys . Para configurar uma conexão criptografada com o broker MQTT, especifique os parâmetros TLS no arquivo de configuração do agente como parâmetros de named session ou default . Atualmente, os parâmetros TLS não podem ser passados como parâmetros de chave de item.
Modbus	Lê dados Modbus.	modbus.get	
MQTT	Recebe valores publicados de tópicos MQTT.	mqtt.get	
MySQL	Monitoramento de MySQL e seus forks.	mysql.custom.query, mysql.db.discovery, mysql.db.size, mysql.get_status_variables, mysql.ping, mysql.replication.discovery, mysql.replication.get_slave_status, mysql.version	Para configurar uma conexão criptografada com o banco de dados, especifique os parâmetros TLS no arquivo de configuração do agente como parâmetros de named session ou default . Atualmente, os parâmetros TLS não podem ser passados como parâmetros de chave de item. A chave do item <code>mysql.custom.query</code> é suportada desde Zabbix 6.0.21.
Netif	Monitoramento de interfaces de rede.	net.if.collisions, net.if.discovery, net.if.in, net.if.out, net.if.total	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .

Nome do plugin	Descrição	Supported item keys	Comentarios
Oracle	Monitoramento do banco de dados Oracle.	oracle.diskgroups.stats, ora-cle.diskgroups.discovery, oracle.archive.info, oracle.archive.discovery, oracle.cdb.info, oracle.custom.query, oracle.datafiles.stats, oracle.db.discovery, oracle.fra.stats, oracle.instance.info, oracle.pdb.info, oracle.pdb.discovery, oracle.pga.stats, oracle.ping, oracle.proc.stats, oracle.redolog.info, oracle.sga.stats, oracle.sessions.stats, oracle.sys.metrics, oracle.sys.params, oracle.ts.stats, oracle.ts.discovery, oracle.user.info, oracle.version	Instale o Oracle Instant Client antes de iniciar o plugin.
Proc	Percentual de utilização da CPU do processo.	proc.cpu.util	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix key .
Redis	Monitoramento do servidor Redis.	redis.config, redis.info, redis.ping, redis.slowlog.count	
Smart	Monitoramento S.M.A.R.T.	smart.attribute.discovery, smart.disk.discovery, smart.disk.get	Direitos de acesso sudo/root ao smartctl são necessários para o usuário que executa o Zabbix agent 2. A versão mínima do smartctl necessária é 7.1.
Swap	Tamanho do espaço de swap em bytes/porcentagem.	system.swap.size	As chaves keys suportadas podem ser usadas com o Zabbix agent 2 apenas no Linux/Windows, tanto como verificação passiva quanto ativa: Configuration parameters
SystemRun	Executa comando especificado.	system.run	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix key .
Systemd	Monitoramento de serviços systemd.	systemd.unit.discovery, systemd.unit.get, systemd.unit.info	Consulte também: Parâmetros de configuração do Plugin Unix/Windows)
TCP	Verificação de disponibilidade de conexão TCP.	net.tcp.port	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix key .
UDP	Monitoramento da disponibilidade e desempenho de serviços UDP.	net.udp.service, net.udp.service.perf	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .

Nome do plugin	Descrição	Supported item keys	Comentarios
Uname	Recuperação de informações sobre o sistema.	system.hostname, system.sw.arch, system.username	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
Uptime	Coleta de métricas de tempo de atividade do sistema.	system.uptime	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix key .
VFSDev	Coleta de métricas do VFS.	vfs.dev.discovery, vfs.dev.read, vfs.dev.write	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
WebCertificate	Monitoramento de certificados de site TLS/SSL.	web.certificate.get	
WebPage	Monitoramento da página Web.	web.page.get, web.page.perf, web.page.regex	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
ZabbixAsync	Coleta assíncrona de métricas.	net.tcp.listen, net.udp.listen, sensor, system.boottime, system.cpu.intr, system.cpu.load, system.cpu.switches, system.hw.cpu, system.hw.macaddr, system.localtime, system.sw.os, system.swap.in, system.swap.out, vfs.fs.discovery	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
ZabbixStats	Métricas internas do servidor/proxy do Zabbix ou número de itens atrasados na fila.	zabbix.stats	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .
ZabbixSync	Coleta síncrona de métricas.	net.dns, net.dns.record, net.tcp.service, net.tcp.service.perf, proc.mem, proc.num, system.hw.chassis, system.hw.devices, system.sw.packages, system.users.num, vfs.dir.count, vfs.dir.size, vfs.fs.get, vfs.fs.inode, vfs.fs.size, vm.memory.size.	As chaves suportadas têm os mesmos parâmetros que as chaves do agente Zabbix keys .

Carregável

Note:

Plugins carregáveis, quando iniciados com: `
 -V --version` -exibe a versão do plugin e informações de licença; `
 -h --help` - exibe informações de ajuda.

Clique no nome do plugin para acessar o repositório do plugin com informações adicionais.

Nome do Plugin	Descrição	Chaves de item suportadas	Comentários
MongoDB	Monitoramento de servidores e clusters MongoDB (banco de dados distribuído baseado em documentos).	mongodb.collection.stats, mon-godb.collections.discovery, mon-godb.collections.usage, mon-godb.connpool.stats, mongodb.db.stats, mon-godb.db.discovery, mon-godb.jumbo_chunks.count, mongodb.oplog.stats, mongodb.ping, mongodb.rs.config, mongodb.rs.status, mon-godb.server.status, mongodb.sh.discovery, mongodb.version	<p>Este plugin é carregável desde o Zabbix 6.0.6 (built-in previously). (integrado anteriormente). Binários pré-compilados do plugin para Windows estão disponíveis desde o Zabbix 6.0.19 e são compatíveis com versões anteriores do 6.0.</p> <p>Para configurar conexões criptografadas com o banco de dados, especifique os parâmetros TLS no arquivo de configuração do agente como parâmetros de named session.</p> <p>Suportado nas versões do plugin 1.2.0, 6.0.13 e mais recentes ¹.</p> <p>Atualmente, os parâmetros TLS não podem ser passados como parâmetros de chave de item.</p> <p>Consulte também MongoDB plugin configuration parameters.</p>
MSSQL	Monitoramento do banco de dados MSSQL.	mssql.availability.group.get, mssql.custom.query, mssql.db.get, mssql.job.status.get, mssql.last.backup.get, mssql.local.db.get, mssql.mirroring.get, mssql.nonlocal.db.get, mssql.perfcounter.get, mssql.ping, mssql.quorum.get, mssql.quorum.member.get, mssql.replica.get, mssql.version	<p>Este plugin é suportado desde o Zabbix 6.0.27. Binários pré-compilados do plugin para Windows estão disponíveis desde o Zabbix 6.0.28 e são compatíveis com a versão 6.0.27.</p> <p>Para configurar uma conexão criptografada com o banco de dados, especifique os parâmetros TLS no arquivo de configuração do agente como parâmetros de named session ou default. Atualmente, os parâmetros TLS não podem ser passados como parâmetros de chave de item.</p> <p>Consulte também MSSQL plugin configuration parameters.</p>

Nome do Plugin	Descrição	Chaves de item suportadas	Comentários
PostgreSQL	Monitoramento de PostgreSQL e forks.	pgsql.autovacuum.count, pgsql.archive, pgsql.bgwriter, pgsql.cache.hit, pgsql.connections, pgsql.custom.query, pgsql.dbstat, pgsql.dbstat.sum, pgsql.db.age, pgsql.db.bloating_tables, pgsql.db.discovery, pgsql.db.size, pgsql.locks, pgsql.oldest.xid, pgsql.ping, pgsql.queries, pgsql.replication.count, pgsql.replication.process, pgsql.replication.process.discovery, pgsql.replication.recovery_role, pgsql.replication.status, pgsql.replication_lag.b, pgsql.replication_lag.sec, pgsql.uptime, pgsql.version, pgsql.wal.stat	Este plugin está disponível desde o Zabbix 6.0.10 (integrado anteriormente). Binários pré-compilados do plugin para Windows estão disponíveis desde Zabbix 6.0.19 e são compatíveis com versões anteriores do 6.0. Para configurar conexões criptografadas com o banco de dados, especifique os parâmetros TLS no arquivo de configuração do agente como parâmetros de named session ou default . Atualmente, os parâmetros TLS não podem ser passados como parâmetros de chave de item. Consulte também PostgreSQL plugin configuration parameters .

Consulte também: [Building loadable plugins](#).

Notas de rodapé

¹ - Desde o Zabbix 6.0.13, os plugins carregáveis começaram a usar o mesmo sistema de versionamento que o próprio Zabbix. Como resultado, a versão do plugin MongoDB plugin mudou de 1.2.0 para 6.0.13 e a versão PostgreSQL plugin mudou de 1.2.1 para 6.0.13.

1 Construindo plugins carregáveis

Visão geral

Esta página fornece os passos necessários para construir um arquivo binário de plugin carregável partir das fontes.

Se o arquivoTAR for baixado, é possível construir o plugin offline, i.e. sem conexão com a internet.

O plugin PostgreSQL é usado como exemplo. Outros plugins carregáveis podem ser construídos de forma similar.

Etapas

1. Baixe os arquivos de plugin em [Zabbix Cloud Images and Appliances](#). A página oficial para download estará disponível em breve.

2. Transfira o arquivo para a máquina onde você está construindo o plugin.

3. Descompactar o arquivo TAR, e.g.:

```
tar xvf zabbix-agent2-plugin-postgresql-6.0.13.tar.gz
```

Certifique-se de substituir "zabbix-agent2-plugin-postgresql-6.0.13.tar.gz" com o nome do arquivo baixado.

4. Acesse o diretório extraído:

```
cd <path to directory>
```

5. Run:

```
make
```


6. O plugin executado pode ser colocado em qualquer lugar contanto que seja carregável pelo Zabbix agent 2. Especifique o caminho para o arquivo binário do plugin no arquivo de configuração do plugin, e.g. em `postgresql.conf` para o plugin PostgreSQL:

```
Plugins.PostgreSQL.System.Path=/path/to/executable/zabbix-agent2-plugin-postgresql
```

7. O caminho para o arquivo de configuração do plugin deve ser especificado no parâmetro 'Include' do arquivo de configuração do Zabbix agent 2:

```
Include=/path/to/plugin/configuration/file/postgresql.conf
```

Alvos do Makefile

Os plugins carregáveis fornecidos pelo Zabbix possuem 'makefiles' simples com os seguintes alvos:

Target	Descrição
make	Construir plugin.
make limpar	Deletar todos os arquivos que são geralmente criados pela construção de plugin.
make verificar	Performar testes próprios. É necessário um banco de dados PostgreSQL real.
make estilo	Verificar código Go style com 'golangci-lint'.
make formatar	Formatar código Go com 'go fmt'.
make distribuição	Criar um arquivo contendo as fontes de plugin e as fontes de todos os pacotes necessários para construir o plugin e seus testes próprios.

3 Triggers

Visão geral

Os gatilhos são expressões lógicas que "avaliam" os dados reunidos por itens e representam o atual estado do sistema.

Enquanto itens são usados para reunir dados de sistema, é altamente pouco prático acompanhar estes dados o tempo todo esperando por uma condição de alarme ou que mereça atenção. O trabalho de "avaliação" dos dados pode ser delegada às expressões de gatilho.

Expressões de gatilho permitem definir um limite de que estado de dado é "aceitável". Portanto, caso um dado de entrada ultrapasse o estado aceitável, um gatilho é "disparado" - ou altera o estado para PROBLEMA.

Um gatilho pode ter os seguintes estados:

VALOR	DESCRIÇÃO
OK	Este é um estado normal do gatilho.
PROBLEMA	Normalmente significa que algo aconteceu. Por exemplo, a carga de processador está muito alta.

Em um gatilho básico nós podemos querer configurar um limite para uma média de cinco minutos de algum dado, por exemplo, a carga de CPU. Isto é realizado pela definição de uma expressão de gatilho onde:

- nós aplicamos a função 'avg' ao valor recebido na chave do item
- nós usamos um período de cinco minutos para avaliação
- nós configuramos um limite de '2'

```
avg(host/key,5m)>2
```

Este gatilho "disparará" (se tornará PROBLEMA) se a média de cinco minutos estiver acima de 2.

Em um gatilho mais complexo, a expressão pode incluir uma **combinação** de múltiplas funções e múltiplos limites. Veja também: [Expressão de gatilho](#).

A maioria das expressões de gatilho são avaliadas com base nos dados **históricos**, enquanto algumas funções de gatilho para análises de longo prazo, p.e. **trendavg()**, **trendcount()**, etc, usam dados estatísticos.

Note:

Após ativar um trigger (alterando seu status de configuração de Desabilitado para Habilitado), a expressão do trigger é avaliada assim que um item nela recebe um valor ou assim que o momento de lidar com a função time-based chegue.

A maioria das funções de trigger são avaliadas com base nos dados de histórico do item **history**, enquanto algumas funções de trigger para análise de longa prazo, como **trendavg()**, **trendcount()**, etc, utilizam dados de tendência.

Tempo de cálculo

Um trigger é recalculado toda vez que o servidor do Zabbix recebe um valor novo que é parte da expressão. Quando um valor novo é recebido pelo servidor, cada função, que está incluída na expressão, é recalculada (não apenas aquela que recebe um valor novo). Além disso, um trigger é recalculado toda vez quando um valor novo é recebido e a cada 30 segundos, se as funções time-based são usadas na expressão. As funções Time-based são **semdado()**, **data()**, **diadomês()**, **diadasemana()**, **hora()**, **agora()**. Essas funções são recalculadas a cada 30 segundos pelo processo de sincronização de histórico do Zabbix. Triggers que referenciam funções trend serão avaliados **apenas** uma vez no menor período de tempo na expressão. Ver também **trend functions**.

Período de avaliação

Um período de avaliação é usado em funções que referenciam o histórico de um item. Ele permite especificar o intervalo no qual estamos interessados. Ele pode ser especificado como um período de tempo (30s, 10m, 1h) ou como um intervalo de valores (#5 - para os últimos cinco valores).

O período de avaliação é medido até "agora" - onde "agora" é o último tempo de recálculo do gatilho (veja **Cálculo de tempo** acima); "agora" não é o horário de "agora" do servidor.

O período de avaliação especifica um de:

- Considera todos os valores entre "agora-período de tempo" e "agora" (ou, com mudança de tempo (time shift), entre "agora-mudança de tempo-período de tempo" e "agora-mudança_tempo")
- Considera não mais do que o número de contagem de valores do passado, até "agora"
 - Se houver 0 valores disponíveis para o período de tempo ou o número de contagem especificado - então o gatilho ou o item calculado que usa esta função se torna não suportado

Note que:

- Se apenas uma única função (referenciando dado histórico) é usada no gatilho, "agora" é sempre o último valor recebido. Por exemplo, se o último valor foi recebido há uma hora atrás, o período de avaliação será considerado em relação ao último valor uma hora atrás.
- Um novo gatilho é calculado tão logo o primeiro valor é recebido (funções históricas); ele será recalculado dentro de 30 segundos para funções baseadas em tempo. Assim, o gatilho será calculado mesmo que de repente o período de avaliação configurado (por exemplo, uma hora) ainda não tenha passado desde a criação do gatilho. O gatilho também será calculado depois do primeiro valor, mesmo que o intervalo de avaliação tenha sido configurado, por exemplo, para os dez últimos valores.

Status desconhecido

É possível que um operando desconhecido apareça em uma expressão de trigger se:

- um item não suportado for utilizado
- a avaliação da função para um item suportado resultar em um erro

Nesse caso, um trigger geralmente é avaliado como "desconhecido" (embora haja algumas exceções). Para mais detalhes, consulte **Expressions with unknown operands**.

É possível receber notificações **get notified** sobre triggers desconhecidos.

1 Configurando um gatilho

Visão geral

Para configurar um gatilho, faça o seguinte:

- Vá até: Configuração → Hosts
- Clique em Gatilhos na linha do host
- Clique em Criar gatilho à direita (ou no nome do gatilho para editar um gatilho existente)
- Informe os parâmetros do gatilho no formulário

Consulte também **informações gerais** sobre gatilhos e seus cálculos de tempo.

Configuração

A aba **Gatilho** contém todos os atributos essenciais de um gatilho.

Trigger
Tags
Dependencies 1

* Name

High CPU utilization (over {SCPU.UTIL.CRIT}% for 5m)

Event name

High CPU utilization (over {SCPU.UTIL.CRIT}% for 5m)

Operational data

Current utilization: {ITEM.LASTVALUE1}

Severity

Not classified

Information

Warning

Average

High

Dis

* Expression

min(/New host/system.cpu.util,5m)>{SCPU.UTIL.CRIT}

Expression constructor

OK event generation

Expression

Recovery expression

None

PROBLEM event generation mode

Single

Multiple

OK event closes

All problems

All problems if tag values match

* Tag for matching

Allow manual close

☐

URL

Description

CPU utilization is too high. The system might be slow to respond.

Enabled

☒

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	<p>Nome do gatilho.</p> <p>Macros suportadas são: {HOST.HOST}, {HOST.NAME}, {HOST.PORT}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {ITEM.VALUE}, {ITEM.LASTVALUE}, {ITEM.LOG.*} e macros de usuário {MACRO}.</p> <p>Macros \$1, \$2...\$9 podem ser usadas para se referir à primeira, segunda...nona constante da expressão.</p> <p>Nota: macros \$1-\$9 resolverão corretamente se referenciando constantes em expressões relativamente simples e diretas. Por exemplo, o nome "Carga processador acima de \$1 em {HOST.NAME}" será automaticamente alterada para "Carga processador acima de 5 em New host" se a expressão é last(/New host/system.cpu.load[percpu,avg1])>5</p>

Parâmetro	Descrição
Nome do evento	<p>Se definido, este nome será usado para criar o nome do evento problema, em vez do nome do gatilho.</p> <p>O nome do evento pode ser usado para construir alertas significantes contendo dados do problema (consulte exemplo).</p> <p>O mesmo conjunto de macros é suportado como no nome do gatilho, mais as macros de expressão {TIME} e {?EXPRESSION}.</p> <p>Suportado desde o Zabbix 5.2.0.</p>
Dados operacionais	<p>Dados operacionais permitem definir strings arbitrárias em conjunto com as macros. As macros resolverão dinamicamente para dados em tempo real em Monitoramento → Problemas.</p> <p>Enquanto macros no nome do gatilho (veja acima) resolverá aos seus valores no momento de ocorrência de um problema e se tornará a base de um nome de problema estático, as macros nos dados operacionais mantêm a habilidade de apresentar a informação mais recente dinamicamente.</p> <p>O mesmo conjunto de macros é suportado como no nome do gatilho.</p>
Severidade	Configure a severidade requerida para o gatilho clicando nos botões.
Expressão	<p>Expressão lógica usada para definir as condições de um problema.</p> <p>Um problema é criado depois que todas as condições inclusas na expressão são atendidas, p.e. a expressão avalia para VERDADEIRO (TRUE). O problema será resolvido tão logo a expressão avalie para FALSO (FALSE), a menos que condições de recuperação adicionais sejam especificadas na Expressão de recuperação.</p>
Geração de evento OK	<p>Opções de geração de evento OK:</p> <p>Expressão - eventos OK são gerados baseados na mesma expressão que os eventos de problema;</p> <p>Expressão de recuperação - eventos OK são gerados se a expressão de problema avaliar para FALSO e a expressão de recuperação avaliar para VERDADEIRO;</p> <p>Nenhum - neste caso o gatilho nunca retornará para um evento OK por conta própria.</p>
Expressão de recuperação	<p>Expressão lógica (opcional) definindo condições adicionais que devem ser atendidas antes do problema ser resolvido, depois que a expressão de problema original já tenha sido avaliada para FALSO.</p> <p>A expressão de recuperação é útil para histerese de gatilho. Não é possível resolver um problema através de uma expressão de recuperação se a expressão de problema ainda for VERDADEIRA.</p> <p>Este campo só está disponível se 'Expressão de recuperação' estiver selecionada para Geração de evento OK.</p>
Modo de geração de evento PROBLEMA	<p>Modo para geração de eventos de problema:</p> <p>Único - um evento único é gerado quando um gatilho entra em estado de 'Problema' pela primeira vez;</p> <p>Múltiplo - um evento é gerado a cada avaliação de 'Problema' no gatilho.</p>
Fechamento por evento OK	<p>Selecione se o evento OK fecha:</p> <p>Todos os problemas - todos os problemas deste gatilho</p> <p>Todos os problemas se valores de etiqueta corresponderem - apenas aqueles problemas do gatilho com valores de etiqueta de evento correspondentes</p>
Etiqueta para correspondência	<p>Informe o nome da etiqueta de evento para usar na correlação de evento.</p> <p>Este campo é apresentado se 'Todos os problemas se valores de etiqueta corresponderem' estiver selecionado para a propriedade Fechamento por evento OK e é obrigatório neste caso.</p>
Permitir fechamento manual	Selecione para permitir fechamento manual de eventos de problema gerados por este gatilho. Fechamento manual é possível quando reconhecendo eventos de problema.

Parâmetro	Descrição
URL	Se não estiver vazio, a URL informada aqui está disponível como um link em vários locais do Frontend, p.e. quando clicando no nome do problema em Monitoramento → Problemas (opção URL no menu do Gatilho) e widget de dashboard Problemas. O mesmo conjunto de macros é suportado assim como no nome do gatilho, mais {EVENT.ID}, {HOST.ID} e {TRIGGER.ID}. Note que macros de usuário com valores secretos não serão resolvidas na URL.
Descrição	Campo de texto usado para fornecer mais informações sobre este gatilho. Pode conter instruções para correção de problemas específicos, detalhes de contato do pessoal responsável, etc. O mesmo conjunto de macros é suportado assim como no nome do gatilho.
Habilitado	Desmarcar esta caixa desabilitará o gatilho se necessário. Problemas de um gatilho desabilitado não são mais apresentados no Frontend, mas não são excluídas.

A aba de **Etiquetas** permite definir **etiquetas** de nível de gatilho. Todos os problemas deste gatilho serão etiquetados com os valores informados aqui.

The screenshot shows the 'Tags' configuration interface. It has two tabs: 'Trigger tags' (active) and 'Inherited and trigger tags'. Below the tabs is a table with the following data:

Name	Value	Action	Parent
App	MySQL	Remove	Templa
tag	value	Remove	

At the bottom left, there is an 'Add' link.

Em adição a opção Etiquetas herdadas e de gatilho permite visualizar etiquetas definidas no nível de template, se o gatilho vem daquele template. Se houver múltiplos templates com a mesma etiqueta, estas etiquetas são apresentadas uma vez e os nomes de template são separados com vírgulas. Um gatilho não "herda" e não apresenta etiquetas de nível de host.

Parâmetro	Descrição
Nome/Valor	Configure etiquetas customizadas para marcar eventos do gatilho. Etiquetas são um par de nome e valor de etiqueta. Você pode usar apenas o nome ou pareá-lo com um valor. Um gatilho pode ter várias etiquetas com o mesmo nome, mas valores diferentes. Macros de usuário, contexto de macro de usuário, macros de descoberta de baixo-nível e funções de macro com {{ITEM.VALUE}}, {{ITEM.LASTVALUE}} e são suportadas em etiquetas de evento. Macros de descoberta de baixo-nível pode ser usadas dentro de contexto de macro. A macro {TRIGGER.ID} é suportada nos valores de etiqueta de gatilho. Ela pode ser útil para identificação de gatilhos criados a partir de protótipos de gatilho e, por exemplo, suprimindo problemas destes gatilhos durante uma manutenção. Se o comprimento total de valores expandidos exceder 255, ele será cortado para 255 caracteres. Veja todas as macros suportadas para etiquetas de evento. Etiquetas de evento podem ser usadas para correlação de evento, em condições de ação e também serão vistas em Monitoramento → Problemas ou no widget de Problemas.

A aba **Dependências** contém todas as **dependências** do gatilho.

Clique em Adicionar para adicionar uma nova dependência.

Note:

Você também pode configurar um gatilho abrindo um existente, pressionando o botão Clonar e então salvando-o sob um nome diferente.

Teste de expressões

É possível testar as expressões de gatilho configuradas para saber qual seria o resultado da expressão dependendo do valor recebido.

A seguinte expressão de um template oficial é tomada como exemplo:

```
avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)>{$TEMP_WARN}  
ou  
last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])={$TEMP_WARN_STATUS}
```

Para testar a expressão, clique no Construtor de expressão sob o campo expressão.

Trigger Tags Dependencies

* Name Cisco IOS SNMPv2: Temperature is too high

Event name Cisco IOS SNMPv2: Temperature is too high

Operational data

Severity Not classified Information **Warning** Average High Disaster

* Expression

```
avg(/Cisco IOS  
SNMPv2/sensor.temp.value[ciscoEnvMonTemperature  
Value.{#SNMPINDEX}],5m)>{$TEMP_WARN}  
or  
last(/Cisco IOS  
SNMPv2/sensor.temp.status[ciscoEnvMonTemperatur  
eState.{#SNMPINDEX}])={$TEMP_WARN_STATUS}
```

Add

[Expression constructor](#)

No construtor de expressão, todas as expressões individuais são listadas. Para abrir a janela de testes, clique em Testar abaixo da lista de expressões.

Target Expression

☒ Or

☐ A avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)>{\$TEMP_WARN}

☐ B last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])={\$TEMP_WARN_STATUS}

[Test](#)

Na janela de testes você pode informar amostras de valor ('80', '70', '0', '1' neste exemplo) e então ver o resultado da expressão, clicando no botão Testar.

Test

Test data

Expression Variable Elements	Result type	Value
avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)	Numeric (float)	80
{\$TEMP_WARN}	Any	70
last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])	Numeric (integer)	0
{\$TEMP_WARN_STATUS}	Any	1

Result

Expression	Result	Error
Or	TRUE	
A avg(/Cisco IOS SNMPv2/sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}],5m)	TRUE	
B last(/Cisco IOS SNMPv2/sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}])	FALSE	
A or B	TRUE	

Test

Cancel

O resultado das expressões individuais assim como da expressão completa podem ser vistas.

"VERDADEIRO" (TRUE) significa que a expressão especificada está correta. Neste caso particular A, "80" é maior que o valor {\$TEMP_WARN} especificado, "70" neste exemplo. Como esperado, um resultado "VERDADEIRO" aparece.

"FALSO" (FALSE) significa que a expressão especificada está incorreta. Neste caso particular B, {\$TEMP_WARN_STATUS} "1" precisa ser igual ao valor especificado, "0" neste exemplo. Como esperado, um resultado "FALSO" aparece.

O tipo de expressão escolhido é "OR". Se ao menos uma das condições especificadas (A ou B neste caso) é VERDADEIRA, o resultado geral será VERDADEIRO também. Significando que o valor atual excede o valor de alerta e um problema ocorreu.

2 Expressões de gatilho

Visão geral

As expressões de triggers são muito flexíveis. Você pode utiliza-las para criar testes com lógicas complexas em relação a dados e estatísticas da monitoração.

A sintaxe básica de uma expressão pode ser definida assim:

```
{<server>:<key>.<function>(<parameter>)}<operator><constant>
```

1 Funções

As funções de Trigger permitem referenciar dados coletados, horário atual e outros fatores.

Uma lista completa das **funções suportadas** está disponível.

2 Parâmetros das funções

A maioria das funções numéricas aceita uma quantidade de segundos como parâmetro (uma unidade de tempo).

Você pode utilizar o prefixo # para especificar um parâmetro com significado diferente:

Chamada da Função	Significa
sum(600)	Sumarização de todos os valores nos últimos 600 segundos (10 minutos)
sum(#5)	Sumarização dos últimos 5 valores

A função **last** utiliza uma forma diferente para valores prefixados com #, para ela isso indicará que ela deverá recuperar o enézimo valor anterior (do mais recente para o mais antigo). Exemplo, suponhamos que os últimos 10 valores coletados são: (2, 3, 4, 9, 10, 11, 11, 16, 7, 0)

- **last(#2)** - irá retornar o penúltimo valor 7
- **last(#5)** - irá retornar 10.

As funções **avg**, **count**, **last**, **min** e **max** possuem um parâmetro adicional: o **time_shift**. Este parâmetro permite referenciar o dado em determinado período de tempo no passado. Por exemplo, **avg(1h,1d)** irá retornar o valor médio analisando 1 hora de valores de 1 dia antes do momento da coleta.

Note:

As triggers só analisam os dados que estão no histórico. Se o dado desejado não estiver no histórico, mas estiver nas médias, a informação das médias **não será utilizada**. Logo, é necessário que se mantenha o histórico por tempo compatível com as triggers que se deseja criar.

Você pode utilizar os **símbolos de unidades** nas expressões, por exemplo '5m' (minutos) ao invés de '300' segundos ou '1d' (dia) ao invés de '86400' segundos, '1K' ao invés de '1024' bytes.

3 Operadores

Os seguintes operadores são suportados nas expressões de triggers (**em ordem decedente de prioridade de execução**):

Prioridade	Operador	Definição
1	-	Símbolo de negativo
2	not	Não Lógico
3	*** Multiplificação /**	Divisão
4	+	Soma aritmética
	-	Redução aritmética
5	<	Menor que. O operador é definido como: $A < B \Leftrightarrow (A <= B - 0.000001)$
	<=	Menor ou igual a.
	>	Maior que. O operador é definido como: $A > B \Leftrightarrow (A >= B + 0.000001)$
	>=	Maior ou igual a.
6	=	É igual. O operador é definido como: $A = B \Leftrightarrow (A > B - 0.000001) \text{ e } (A < B + 0.000001)$
	<>	Diferente. O operador é definido comoas: $A <> B \Leftrightarrow (A <= B - 0.000001) \text{ ou } (A >= B + 0.000001)$
7	and	Operador lógico E
8	or	Operador lógico OU

Os operadores **not**, **and** e **or** são sensíveis ao caso e deverão ser escritos em letra minúscula. Eles devem estar envoltos em espaços ou parênteses.

Todos os operadores, exceto o "Símbolo de negativo" e "Não lógico", tem associação da esquerda para a direita.

4 Exemplos de triggers

Exemplo 1

Análise de carga de processamento na CPU: "Processor load is too high on www.zabbix.com"

```
{www.zabbix.com:system.cpu.load[all,avg1].last()}>5
```

'www.zabbix.com:system.cpu.load[all,avg1]' provê um nome curto para o parâmetro monitorado. Neste caso se refere ao host 'www.zabbix.com' com a chave 'system.cpu.load[all,avg1]'. Utilizando a função 'last()', nós estaremos referindo ao valor mais recente. Finalmente, '>5' define que a trigger deverá ir para o estado de "INCIDENTE" quando o valor mais recente desta chave, neste host for superior a 5.

Exemplo 2

www.zabbix.com is overloaded

```
{www.zabbix.com:system.cpu.load[all,avg1].last()}>5 ou {www.zabbix.com:system.cpu.load[all,avg1].min(10m)}
```

A expressão será verdadeira quando o último valor coletado para a carga da CPU for superior a 5 ou superior a 2 nos últimos 10 coletados.

Exemplo 3

/etc/passwd has been changed

Utilize a função 'diff()':

```
{www.zabbix.com:vfs.file.cksum[/etc/passwd].diff()}=1
```

A expressão será verdadeira quando o último valor da verificação 'checksum' do arquivo '/etc/passwd' for diferente da penúltima verificação.

De forma similar esta técnica pode ser utilizada para monitorar vários outros arquivos, tais quais: /etc/inetd.conf, /kernel, etc.

Exemplo 4

Alguém está baixando um arquivo muito grande da internet (ou um tráfego intenso por um longo período)

Utilize a função 'min()':

```
{www.zabbix.com:net.if.in[eth0,bytes].min(5m)}>100K
```

A expressão será verdadeira quando a quantidade de bytes recebidos nos últimos 5 minutos na interface 'eth0' for superior a 100 KB.

Exemplo 5

Ambos os nós do cluster de SMTP estão indisponíveis

Observe que a expressão utiliza dados de dois hosts diferentes:

```
{smtp1.zabbix.com:net.tcp.service[smtp].last()}=0 and {smtp2.zabbix.com:net.tcp.service[smtp].last()}=0
```

A expressão será verdadeira quando ambos os servidores (smtp1.zabbix.com e smtp2.zabbix.com) SMTP estiverem fora do ar.

Exemplo 6

A versão do Zabbix Agent precisa ser atualizada

Use a função 'str()':

```
{zabbix.zabbix.com:agent.version.str("beta8")}=1
```

A expressão será verdadeira se a versão do Zabbix Agent possuir o texto "beta8" (por exemplo 1.0beta8).

Exemplo 7

Servidor indisponível

```
{zabbix.zabbix.com:icmping.count(30m,0)}>5
```

A expressão será verdadeira se o host "zabbix.zabbix.com" estiver inacessível por mais de 5 vezes nos últimos 30 minutos.

Exemplo 8

Sem dados nos últimos 3 minutos

Use a função 'nodata()':

```
{zabbix.zabbix.com:tick.nodata(3m)}=1
```

Note:

Neste exemplo 'tick' é um item do tipo 'Zabbix trapper'. Para que esta trigger funcione o item 'tick' precisará ter sido definido. O host precisará enviar periodicamente o dado para este item através do comando 'zabbix_sender' ou similar.

A expressão será verdadeira se nenhum dado for recebido nos últimos 180 segundos.

Exemplo 9

Alta carga de CPU no período noturno

Utilize a função 'time()':

```
{zabbix:system.cpu.load[all,avg1].min(5m)}>2 and {zabbix:system.cpu.load[all,avg1].time()}>000000 and {zabbix:system.cpu.load[all,avg1].time()}<000000
```

A expressão será verdadeira se a carga de CPU for superior a 2, entre a meia noite e as seis da manhã.

Exemplo 10

Verifica se o horário local do host monitorado e do servidor do Zabbix estão sincronizados

Use a função 'fuzzytime()':

```
{MySQL_DB:system.localtime.fuzzytime(10)}=0
```

A expressão será verdadeira se o horário do servidor 'MySQL_DB' tiver uma diferença maior que 10 segundos em relação ao horário do Zabbix Server.

Exemplo 11

Comparando a carga atual de CPU com a carga no mesmo horário do dia anterior (usando o parâmetro de time_shift).

```
{server:system.cpu.load.avg(1h)}/{server:system.cpu.load.avg(1h,1d)}>2
```

A expressão será verdadeira se a carga da última hora for duas vezes superior a carga deste mesmo período um dia antes (24 horas).

Exemplo 12

Usando o valor de outro item como limite para a trigger:

```
{Template PfSense:hrStorageFree[{#SNMPVALUE}].last()}<{Template PfSense:hrStorageSize[{#SNMPVALUE}].last() }
```

A expressão será verdadeira se o espaço livre for inferior a 10%.

5 Técnicas 'anti-flapping'

Algumas vezes você precisa ter condições diferentes para estados diferentes (INCIDENTE/OK). Por exemplo, nós podemos ter que definir uma trigger para avisar quando a temperatura de uma sala for superior a 20C (vinte graus) que é o máximo suportável para os servidores funcionarem com segurança, mas a temperatura ideal de funcionamento deveria ser de até 15C (quinze graus). Temos como definir uma trigger desta forma no Zabbix, ela será ativada (mudar para o estado de INCIDENTE) se a temperatura ultrapassar o máximo aceitável, mas não será inativada (retornar ao estado OK) enquanto a temperatura não for inferior à temperatura ideal.

Para fazer isso podemos definir uma trigger como a do "Exemplo 1". A trigger do "Exemplo 2" apresenta a mesma técnica de "anti-flapping" para espaço em disco.

Example 10

CPU activity at any time with exception.

Use of function time() and **not** operator:

```
min(/zabbix/system.cpu.load[all,avg1],5m)>2  
and not (dayofweek()=7 and time()>230000)  
and not (dayofweek()=1 and time()<010000)
```

The trigger may change its state to true at any time, except for 2 hours on a week change (Sunday, 23:00 - Monday, 01:00).

Exemplo 1

A temperatura na sala dos servidores está muito alta

```
({TRIGGER.VALUE}=0 and {server:temp.last()}>20) or  
({TRIGGER.VALUE}=1 and {server:temp.last()}>15)
```

Exemplo 2

Pouco espaço livre no disco

Incidente: se for menor que 10GB nos últimos 5 minutos

Recuperação (OK): se for maior que 40GB nos últimos 10 minutos

```
({TRIGGER.VALUE}=0 and {server:vfs.fs.size[/,free].max(5m)}<10G) or  
({TRIGGER.VALUE}=1 and {server:vfs.fs.size[/,free].min(10m)}<40G)
```

Note:

Observe que a macro {TRIGGER.VALUE} retorna o estado corrente da trigger (0 - OK, 1 - Incidente).

Example 2

Free disk space is too low.

Problem expression: it is less than 10GB for last 5 minutes

```
max(/server/vfs.fs.size[/,free],5m)<10G
```

Recovery expression: it is more than 40GB for last 10 minutes

```
min(/server/vfs.fs.size[/,free],10m)>40G
```

Expressions with unsupported items and unknown values

Versions before Zabbix 3.2 are very strict about unsupported items in a trigger expression. Any unsupported item in the expression immediately renders trigger value to **Unknown**.

Since Zabbix 3.2 there is a more flexible approach to unsupported items by admitting unknown values into expression evaluation:

- For the nodata() function, the values are not affected by whether an item is supported or unsupported. The function is evaluated even if it refers to an unsupported item.

- Logical expressions with OR and AND can be evaluated to known values in two cases regardless of unknown operands:
 - "1 or Unsuported_item1.some_function() or Unsuported_item2.some_function() or ..." can be evaluated to '1' (True),
 - "0 and Unsuported_item1.some_function() and Unsuported_item2.some_function() and ..." can be evaluated to '0' (False).
 Zabbix tries to evaluate logical expressions taking unsupported items as `Unknown` values. In the two cases mentioned above a known value will be produced; in other cases trigger value will be `Unknown`.
- If a function evaluation for supported item results in error, the function value is `Unknown` and it takes part in further expression evaluation.

Note that unknown values may "disappear" only in logical expressions as described above. In arithmetic expressions unknown values always lead to result `Unknown` (except division by 0).

If a trigger expression with several unsupported items evaluates to `Unknown` the error message in the frontend refers to the last unsupported item evaluated.

Example 12

Using the value of another item to get a trigger threshold:

```
last(/Template PfSense/hrStorageFree[{#SNMPVALUE}])<last(/Template PfSense/hrStorageSize[{#SNMPVALUE}])*0.1
```

The trigger will fire if the free storage drops below 10 percent.

Example 13

Using **evaluation result** to get the number of triggers over a threshold:

```
(last(/server1/system.cpu.load[all,avg1])>5) + (last(/server2/system.cpu.load[all,avg1])>5) + (last(/server3/system.cpu.load[all,avg1])>5)
```

The trigger will fire if at least two of the triggers in the expression are over 5.

Example 14

Comparing string values of two items - operands here are functions that return strings.

Problem: create an alert if Ubuntu version is different on different hosts

```
last(/NY Zabbix server/vfs.file.contents[/etc/os-release])<>last(/LA Zabbix server/vfs.file.contents[/etc/os-release])
```

Example 15

Comparing two string values - operands are:

- a function that returns a string
- a combination of macros and strings

Problem: detect changes in the DNS query

The item key is:

```
net.dns.record[8.8.8.8,{ $WEBSITE_NAME }, { $DNS_RESOURCE_RECORD_TYPE }, 2, 1]
```

with macros defined as

```
{ $WEBSITE_NAME } = example.com
{ $DNS_RESOURCE_RECORD_TYPE } = MX
```

and normally returns:

```
example.com          MX          0 mail.example.com
```

So our trigger expression to detect if the DNS query result deviated from the expected result is:

```
last(/Zabbix server/net.dns.record[8.8.8.8,{ $WEBSITE_NAME }, { $DNS_RESOURCE_RECORD_TYPE }, 2, 1])<>"{ $WEBSITE_NAME } { $DNS_RESOURCE_RECORD_TYPE }"
```

Notice the quotes around the second operand.

Example 16

Comparing two string values - operands are:

- a function that returns a string
- a string constant with special characters \ and "

Problem: detect if the /tmp/hello file content is equal to:

```
\ " //hello ? \"
```

Option 1) write the string directly

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])="\\" //hello ?\\""
```

Notice how \ and " characters are escaped when the string gets compared directly.

Option 2) use a macro

```
{$HELLO_MACRO} = \" //hello ?\"
```

in the expression:

```
last(/Zabbix server/vfs.file.contents[/tmp/hello])={$HELLO_MACRO}
```

Example 17

Comparing long-term periods.

Problem: Load of Exchange server increased by more than 10% last month

```
trendavg(/Exchange/system.cpu.load,1M:now/M)>1.1*trendavg(/Exchange/system.cpu.load,1M:now/M-1M)
```

You may also use the **Event name** field in trigger configuration to build a meaningful alert message, for example to receive something like

```
"Load of Exchange server increased by 24% in July (0.69) comparing to June (0.56)"
```

the event name must be defined as:

```
Load of {HOST.HOST} server increased by {{?100*trendavg(/system.cpu.load,1M:now/M)/trendavg(/system.cpu.
```

It is also useful to allow manual closing in trigger configuration for this kind of problem.

Hysteresis

Sometimes an interval is needed between problem and recovery states, rather than a simple threshold. For example, if we want to define a trigger that reports a problem when server room temperature goes above 20°C and we want it to stay in the problem state until the temperature drops below 15°C, a simple trigger threshold at 20°C will not be enough.

Instead, we need to define a trigger expression for the problem event first (temperature above 20°C). Then we need to define an additional recovery condition (temperature below 15°C). This is done by defining an additional Recovery expression parameter when **defining** a trigger.

In this case, problem recovery will take place in two steps:

- First, the problem expression (temperature above 20°C) will have to evaluate to FALSE
- Second, the recovery expression (temperature below 15°C) will have to evaluate to TRUE

The recovery expression will be evaluated only when the problem event is resolved first.

Warning:

The recovery expression being TRUE alone does not resolve a problem if the problem expression is still TRUE!

Example 1

Temperature in server room is too high.

Problem expression:

```
last(/server/temp)>20
```

Recovery expression:

```
last(/server/temp)<=15
```

3 Dependências dos gatilhos

Visão geral

Algumas vezes a disponibilidade de um host depende de outro. Um servidor que está atrás de algum roteador se tornará indisponível se o roteador cair. Com gatilhos configurados para ambos, você pode obter notificações sobre dois hosts fora - enquanto apenas o roteador é a parte culpada.

Aqui é onde alguma dependência entre hosts pode ser útil. Com dependência configurada as notificações dos dependentes podem ser retidas e apenas as notificações para o problema raiz enviadas.

Enquanto o Zabbix não suporta dependência entre hosts diretamente, elas podem ser definidas com outro método, mais flexível - dependências de gatilho. Um gatilho pode ter um ou mais gatilhos dos quais ele depende.

Então em nosso simples exemplo nós abrimos o formulário de configuração de gatilho do servidor e informamos que ele depende do respectivo gatilho do roteador. Com tal dependência o gatilho do servidor não mudará de estado enquanto o gatilho do qual ele depende estiver no estado de 'PROBLEMA' - e assim nenhuma ação será tomada e nenhuma notificação enviada para o dependente.

Se ambos servidor e roteador estiverem fora e houver dependência entre eles, o Zabbix não executará ações para o gatilho dependente.

Ações nos gatilhos dependentes não serão executadas se o gatilho do qual eles dependem:

- mudar seu estado de 'PROBLEMA' para 'DESCONHECIDO'
- for fechado manualmente, por correlação ou com a ajuda de funções baseadas em tempo
- for resolvido por um valor de um item não envolvido com o gatilho dependente
- estiver desabilitado, possuir item desabilitado ou item de host desabilitado

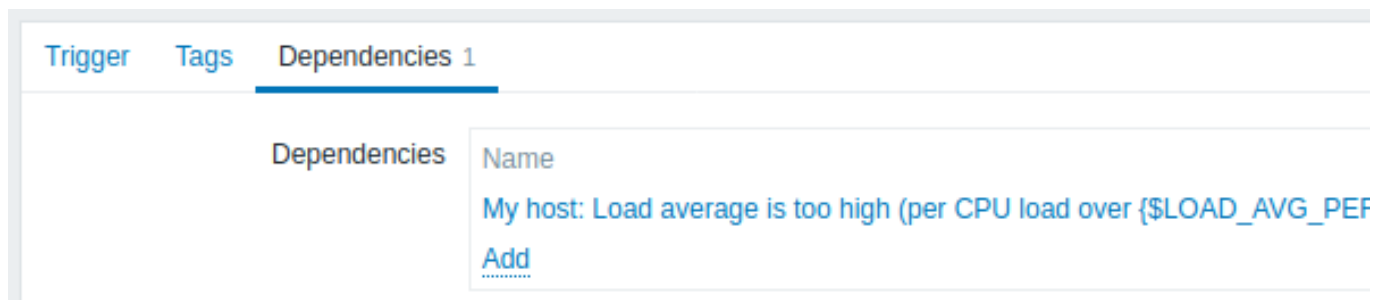
Note que o gatilho "secundário" (dependente) nos casos acima mencionados não serão imediatamente atualizados. Enquanto o gatilho principal estiver em estado de PROBLEMA, seus dependentes podem reportar valores, nos quais não podemos confiar. Assim, o gatilho só será reavaliado, e mudará seu estado, apenas depois que o gatilho principal estiver em estado OK e tivermos recebido métricas confiáveis.

Também:

- Dependência de gatilho pode ser adicionada de qualquer gatilho de host para qualquer gatilho de host, desde que isso não resulte em uma dependência circular.
- Dependência de gatilho pode ser adicionada de um template para outro template. Se um gatilho do template A depende de um gatilho do template B, o template A só poderá ser associado a um host (ou outro template) em conjunto com o template B, mas o template B pode ser associado a um host (ou outro template) de forma isolada.
- Dependência de gatilho pode ser adicionada a partir do gatilho do template para um gatilho de host. Neste caso, a associação de tal template a um host criará um gatilho de host que depende do mesmo gatilho do qual o gatilho do template também depende. Isto permite a, por exemplo, ter um template onde alguns gatilhos dependem dos gatilhos do roteador (host). Todos os hosts associados a este template dependerão deste roteador específico.
- Dependência de gatilho de um gatilho de host para um gatilho de template não pode ser adicionado.
- Dependência de gatilho pode ser adicionada de um protótipo de gatilho para outro protótipo de gatilho (dentro da mesma regra de descoberta de baixo-nível) ou um gatilho real. Um protótipo de gatilho não pode depender de um protótipo de gatilho de uma regra LLD (descoberta de baixo-nível) diferente ou de um gatilho criado a partir de um protótipo de gatilho. Um protótipo de gatilho de host não pode depender de um gatilho de um template.

Configuração

Para definir uma dependência, abra a aba Dependências no **formulário de configuração** do gatilho. Clique em Adicionar no bloco 'Dependências' e selecione um ou mais gatilhos dos quais nosso gatilho dependerá.



Clique em Atualizar. Agora o gatilho tem uma indicação de sua dependência na lista.

Template Module Linux CPU by Zabbix agent: High CPU utilization
(over { \$CPU.UTIL.CRIT }% for 5m)
Depends on:
My host: Load average is too high (per CPU load over
{ \$LOAD_AVG_PER_CPU.MAX.WARN } for 5m)

Exemplo de várias dependências

Por exemplo, um host está atrás de um Roteador2 e o Roteador2 está atrás de um Roteador1.

Zabbix - Roteador1 - Roteador2 - Host

Se o Roteador1 estiver fora, então obviamente Host e Roteador2 estarão também inalcançáveis, ainda assim não queremos receber três notificações sobre Host, Roteador1 e Roteador2 estarem todos fora.

Então neste caso definimos duas dependências:

O gatilho 'Host está fora (down)' depende do gatilho 'Roteador2 está fora'

O gatilho 'Roteador2 está fora' depende do gatilho 'Roteador1 está fora'

Antes de mudar o estado do gatilho 'Host está fora', o Zabbix verificará por dependências de gatilho correspondentes. Se encontradas, e um destes gatilhos estiver em estado de 'PROBLEMA', então o estado do gatilho não será alterado e portanto ações não serão executadas e notificações não serão enviadas.

O Zabbix executa esta verificação recursivamente. Se Roteador1 ou Roteador2 estiver inalcançável, o gatilho do Host não será atualizado.

4 Gravidade do trigger

A gravidade do trigger representa o nível de importância de um trigger.

Severity Not classified Information Warning Average High Disaster

O Zabbix suporta as seguintes gravidades padrão de trigger.

Gravidade	Cor	Descrição
Não classificado	Cinza	Pode ser usado quando o nível de gravidade de um evento é desconhecido, não foi determinado, não faz parte do escopo de monitoramento regular, etc., por exemplo, durante a configuração inicial, como um espaço reservado para avaliações futuras, ou como parte de um processo de integração.
Informação	Azul claro	Pode ser usado para eventos informativos que não exigem atenção imediata, mas ainda podem fornecer informações valiosas.
Aviso	Amarelo	Pode ser usado para indicar um problema potencial que pode exigir investigação ou ação, mas que não é crítico.
Médio	Laranja	Pode ser usado para indicar um problema significativo que deve ser abordado relativamente em breve para evitar problemas adicionais.
Alto	Vermelho claro	Pode ser usado para indicar problemas críticos que precisam de atenção imediata para evitar interrupções significativas.
Desastre	Vermelho	Pode ser usado para indicar um incidente grave que requer ação imediata para evitar, por exemplo, interrupções do sistema ou perda de dados.

Note:

Os nomes e cores das gravidades do trigger podem ser personalizados **customized**.

As gravidades do trigger são usadas para:

- representação visual de triggers - cores diferentes para diferentes gravidades;
- áudio em alarmes globais - áudio diferente para diferentes gravidades;
- mídia do usuário - mídia diferente (canal de notificação) para diferentes gravidades (por exemplo, SMS para triggers de gravidade Alto e Desastre, e E-mail para triggers de outras gravidades);
- limitando ações por condições contra gravidades de trigger.

5 Customizando severidades de gatilho

Os nomes das severidades de gatilho e cores para elementos de GUI relacionados à severidade podem ser configurados em Administração → Geral → Opções de apresentação de gatilho. Tais cores são compartilhadas entre todos os temas de GUI.

Tradução de nomes de severidade customizados

Attention:

Se as traduções do Zabbix Frontend são usadas, nomes de severidade customizados irão sobrescrever os nomes traduzidos por padrão.

Nomes de severidade de gatilho padrão estão disponíveis para tradução em todos as localizações. Se um nome de severidade é alterado, um nome customizado é usado em todas as localizações e tradução manual adicional é necessária.

Procedimento de tradução de nome de severidade customizado:

- informe o nome de severidade customizado requerido, por exemplo, 'Important'
- edite <frontend_dir>/locale/<required_locale>/LC_MESSAGES/frontend.po
- adicione 2 linhas:

```
msgid "Important"
msgstr "<translation string>"
```

e salve o arquivo.

- crie arquivos .mo como descrito em <frontend_dir>/locale/README

Aqui **msgid** deve corresponder ao nome da nova severidade customizada e **msgstr** deve ser a tradução para ele no idioma específico.

Este procedimento deve ser executado depois de cada alteração de nome de severidade.

6 Atualização em massa

Visão geral

Com a atualização em massa, você pode alterar alguns atributos para um número de triggers de uma só vez, evitando que você precise abrir cada trigger individualmente para edição.

Usando atualização em massa

Para atualizar alguns triggers em massa, faça o seguinte:

- Marque as caixas dos triggers que você deseja atualizar na lista
- Clique em Atualização em massa abaixo da lista
- Navegue até a aba com os atributos desejados (Trigger, Tags ou Dependências)
- Marque as caixas dos atributos para atualização

Mass update

[Trigger](#) [Tags](#) [Dependencies](#)

Severity ☒

Not classified

Information

Warning

Average

High

Disaster

Allow manual close ☐ Original

Mass update

[Trigger](#) [Tags](#) [Dependencies](#)

Tags ☒

Add

Replace

Remove

Name

Value

tag

value

[Add](#)

As seguintes opções estão disponíveis ao selecionar o respectivo botão para atualização de tags:

- Adicionar - permite adicionar novas tags para os triggers;
- Substituir - removerá quaisquer tags existentes do trigger e substituirá por aquela(s) especificada(s) abaixo;
- Remover - removerá do trigger as tags especificadas.

Observe que tags com o mesmo nome, mas valores diferentes não são consideradas 'duplicadas' e podem ser adicionadas ao mesmo trigger.

Mass update

Trigger

Tags

Dependencies

Replace dependencies ☒

Name

Zabbix server: Lack of available memory (< 20M of 7.72 GB)

Add

Substituir dependências - removerá quaisquer dependências existentes do trigger e substituirá por aquela(s) especificada(s).

Clique em Atualizar para aplicar as alterações.

7 Funções de triggers preditivas

Visão geral

Algumas vezes há sinais dos problemas que ocorrerão. Estes sinais podem ser identificados de forma que ações possam ser tomadas de antemão para prevenir ou ao menos minimizar os impactos do problema.

O Zabbix possui ferramentas para prever o comportamento futuro de um sistema monitorado baseado em seus dados históricos. Estas ferramentas são disponibilizadas através das funções de gatilho preditivas.

1 Funções

Duas coisas que se precisa conhecer é como definir um estado de problema e quanto tempo é necessário para tomar uma ação. Então há dois caminhos para configurar uma sinalização de gatilho sobre uma situação indesejada em potencial. Primeiro: o gatilho deve disparar quando é esperado que o sistema esteja em estado de problema após o "tempo para agir". Segundo: o gatilho deve disparar quando o sistema alcançará o estado de problema em menos do que o "tempo para agir".

Funções de gatilhos correspondentes para usar são **forecast** e **timeleft**. Note que a análise estatística subjacente é basicamente idêntica para ambas as funções. Você pode configurar um gatilho da maneira que preferir com resultados similares.

2 Parâmetros

Ambas as funções utilizam praticamente o mesmo conjunto de parâmetros. Use a lista de **funções suportadas** para referência.

2.1 Intervalo de tempo

Primeiro de tudo, você deve especificar o período histórico que o Zabbix deve analisar para chegar a uma predição. Você faz isso de uma forma familiar com o parâmetro `período de tempo` e mudança de tempo (time shift) opcional como você fez com as funções **avg**, **count**, **delta**, **max**, **min** e **sum**.

2.2 Horizonte de previsão

(apenas **forecast**)

O parâmetro `time` especifica quanto distante no futuro o Zabbix deve extrapolar as dependência que ele encontra nos dados históricos. Não importa se você usar `time_shift` ou não, `time` é sempre contado a partir momento atual.

2.3 Limite (threshold) a alcançar

(apenas **timeleft**)

O parâmetro `threshold` especifica um valor que o item analisado deve alcançar, sem distinção se a partir de cima ou de baixo. Uma vez que tenhamos determinado $f(t)$ (veja abaixo) devemos solucionar a equação $f(t) = \text{threshold}$ e retornar a raiz que está mais perto e à direita de agora (now) ou 9999999999.9999 se não houver tal raiz.

Note:

Quando valores de item se aproximam do limite e então o cruzam, **timeleft** assume que essa interseção já está no passado e por isso alterna para a próxima interseção com o nível de `threshold`, se houver algum. A boa prática deveria ser utilizar as predições como complemento aos diagnósticos de problema ordinários, e não como uma substituição.^a

^aDe acordo com [especificação](#) estas são voltagens nos pinos do chip e, de um modo geral, podem precisar dimensionamento.

2.4 Funções de ajuste (fit)

O ajuste (fit) padrão é a função linear. Mas se o seu sistema monitorado é mais complexo você tem mais opções para escolher.

fit	$x = f(t)$
linear	$x = a + b*t$
polynomial ¹ (polinomial)	$x = a_0 + a_1*t + a_2*t^2 + \dots + a_n*t^n$
exponential (exponencial)	$x = a*\exp(b*t)$
logarithmic (logarítmico)	$x = a + b*\log(t)$
power (potência)	$x = a*t^b$

2.5 Modos

(apenas **forecast**)

Cada vez que uma função de gatilho é avaliada ela obtém seus dados do período histórico especificado e ajusta uma função especificada para os dados. Então, se os dados forem ligeiramente diferentes a função ajustada será ligeiramente diferente. Se nós simplesmente calcularmos o valor da função ajustada em um momento específico no futuro você não saberá nada sobre como é esperado que o item analisado se comporte entre agora e aquele momento no futuro. Para algumas opções de fit (como polynomial) um simples valor do futuro pode ser enganoso.

modo	resultado forecast
valor	$f(\text{now} + \text{time})$
max	$\max_{\text{now} \leq t \leq \text{now} + \text{time}} f(t)$
min	$\min_{\text{now} \leq t \leq \text{now} + \text{time}} f(t)$
delta	max - min
avg	média de $f(t)$ ($\text{now} \leq t \leq \text{now} + \text{time}$) de acordo com definição

3 Detalhes

Para evitar cálculos com números enormes nós consideramos o registro de data do primeiro valor no período especificado mais 1 ns como um novo tempo zero (zero-time) (tempo epoch atual é de ordem 10^9 , epoch ao quadrado é 10^{18} , dupla precisão é aproximadamente 10^{-16}). 1 ns é adicionado para fornecer todos os valores de tempo positivos para ajustes logarithmic e power que envolvem o cálculo de $\log(t)$. Alteração de tempo (time shift) não afeta as funções linear, polynomial, exponential (além de cálculos mais fáceis e mais precisos) mas altera a forma das funções logarithmic e power.

4 Erros potenciais

As funções retornam -1 em tais situações:

- período de avaliação especificado não contém dados;
- resultado da operação matemática não está definido ²;
- complicações numéricas (infelizmente, para alguns conjuntos de dados de entrada, intervalo e precisão de formatos de ponto flutuante de dupla precisão se tornam insuficientes) ³.

Note:

Nenhum alerta ou erros são apresentados se o ajuste (fit) escolhido descreve pobremente os dados providos ou se há apenas poucos dados para um predição acurada.

5 Exemplos e lidando com erros

Para receber um alerta quando você estiver prestes a ficar sem espaço livre em disco no seu host, você deve usar uma expressão de gatilho como esta:

² Por exemplo o ajuste de funções exponencial ou power envolvem o cálculo de $\log()$ dos valores do item. Se o dado contém zeros ou valores negativos você terá um erro pois $\log()$ é definido apenas para valores positivos.

³ Para ajustes linear, exponencial, logarithmic e power todos os cálculos necessários podem ser escritos explicitamente. Para polynomial apenas valor pode ser calculado sem quaisquer passos adicionais. O cálculo de avg envolve computação de antiderivativo polinomial (analiticamente). Computação de max, min e delta envolve computação de derivativo polinomial (analiticamente) e descoberta de suas raízes (numericamente). A solução de $f(t) = 0$ envolve a descoberta das raízes polinomiais (numericamente).

```
timeleft(/host/vfs.fs.size[/,free],1h,0)}<1h
```

No entanto, o código de erro -1 pode aparecer e colocar seu gatilho em um estado de problema. Geralmente isso é bom porque você recebe um alerta de que suas previsões não funcionam corretamente e você deve olhar para elas com mais cuidado para encontrar o porquê. Mas algumas vezes é ruim porque -1 pode simplesmente significar que não há dados sobre o espaço em disco livre no host obtidos na última hora. Se você estiver recebendo alertas de falso positivo demais considere usar uma expressão de gatilho mais elaborada ⁴:

```
timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1
```

A situação é um pouco mais difícil com **forecast**. Primeiro de tudo, -1 pode ou não colocar o gatilho em um estado de problema dependendo se você tiver uma expressão como `forecast(/host/item,...)<...` ou como `forecast(/host/item,...)>...`.

Além do mais, -1 pode ser uma previsão válida se for normal o valor do item ser negativo. Mas a probabilidade desta situação em uma situação do mundo real é desprezível (veja **como** o operador = funciona). Então adicione `... or forecast(/host/item,...)=-1` ou `... and forecast(/host/item,...)<>-1` se você quiser ou não tratar -1 como um problema, respectivamente.

4 Eventos

Visão geral

Há vários tipos de eventos gerados no Zabbix:

- eventos de gatilho - sempre que um gatilho altera seu estado (OK→PROBLEMA→OK)
- eventos de serviço - sempre que um serviço altera seu estado (OK→PROBLEMA→OK)
- eventos de descoberta - quando hosts ou serviços são detectados
- eventos de autoregistro - quando agentes ativos são autoregistrados pelo Server
- eventos internos - quando um item/regra de descoberta de baixo-nível se torna não suportado ou um gatilho entra em estado desconhecido

Note:

Eventos internos são suportados a partir da versão de Zabbix 2.2.

Eventos possuem carimbo de hora e podem ser a base de ações como envio de e-mail de notificação, etc.

Para ver detalhes de eventos no Frontend, vá até Monitoramento → Problemas. Ali você pode clicar na data e horário do evento para visualizar detalhes de um evento.

Mais informações estão disponíveis em:

- [eventos de gatilho](#)
- [outras fontes de evento](#)

1 Geração de evento de gatilho

Visão geral

Mudança de estado de gatilho é a origem de eventos mais frequente e mais importante. Cada vez que o gatilho muda seu estado, um evento é gerado. O evento contém detalhes da alteração de estado do gatilho - quanto ela ocorreu e qual é o novo estado.

Dois tipos de eventos são criados pelos gatilhos - PROBLEMA e OK.

Eventos PROBLEMA

Um evento problema é criado:

- quando uma expressão de gatilho avalia para VERDADEIRO (TRUE) se o gatilho está em estado OK;
- a cada vez que uma expressão de gatilho avalia para VERDADEIRO se a geração de múltiplos eventos problema está habilitada para o gatilho.

Eventos OK

Um evento OK encerra o(s) evento(s) problema relacionado(s) e pode ser criado por 3 componentes:

⁴ Mas neste caso o -1 pode fazer com que seu gatilho se recupere do estado de problema. Para estar completamente protegido use: `timeleft(/host/vfs.fs.size[/,free],1h,0)<1h and ({TRIGGER.VALUE}=0 and timeleft(/host/vfs.fs.size[/,free],1h,0)<>-1 or {TRIGGER.VALUE}=1)`

- gatilhos - baseado nas configurações 'geração de evento OK' e 'encerramento de evento OK';
- correlação de evento
- gerenciador de tarefa - quando um evento é **encerrado manualmente**

Gatilhos

Os gatilhos tem uma configuração de 'geração de evento OK' que controla como os eventos OK são gerados:

- Expressão - um evento OK é gerado para um gatilho em estado de problema quando sua expressão avalia para FALSO. Esta é a configuração mais simples, habilitada por padrão.
- Expressão de recuperação - um evento OK é gerado para um gatilho em estado problema quando sua expressão avalia para FALSO e a expressão de recuperação avalia para VERDADEIRO. Isto pode ser usado se o critério de recuperação do gatilho é diferente do critério de problema.
- Nenhum - um evento OK nunca é gerado. Isto pode ser usado em conjunto com geração de evento problema múltiplo para simplificar o envio de uma notificação quando algo ocorrer.

Adicionalmente os gatilhos possuem uma configuração de 'Evento OK encerra' que controla quais eventos problema são encerrados:

- Todos os problemas - um evento OK encerrará todos os problemas abertos criados pelo gatilho
- Todos os problemas se valores de etiqueta correspondem - um evento OK encerrará problemas criados pelo gatilho e tendo no mínimo um valor de etiqueta correspondente. A etiqueta é definida pela configuração de gatilho 'Etiqueta para correspondência'. Se não há eventos problema a serem encerrados então o evento OK não é gerado. Isto é frequentemente chamado de correlação de evento em nível de gatilho.

Correlação de evento

Correlação de evento (também chamado de correlação de evento global) é um modo de definir regras customizadas de encerramento de evento (resultando na geração de evento OK).

As regras definem como os novos eventos problema são pareados com eventos problema existentes e permite encerrar o novo evento ou os eventos correspondidos pela geração de eventos OK correspondentes.

No entanto, a correlação de evento deve ser configurada com muito cuidado, pois ela pode afetar negativamente a performance do processamento de evento ou, se mal configurada, encerrar mais eventos do que o esperado (no pior cenário pode até mesmo encerrar todos os eventos). Algumas dicas de configuração:

1. sempre reduza o escopo de correlação pela definição de uma etiqueta única para o evento controle (o evento que é pareado com eventos antigos) e use a condição de correlação 'nova etiqueta de evento'
2. não esqueça de adicionar uma condição baseada no evento antigo quando usando a operação 'encerrar evento antigo', ou todos os problemas existentes podem ser encerrados
3. evite o uso de nomes de etiqueta comuns usados por diferentes configurações de correlação

Gerenciador de tarefa

Se a configuração 'Permitir encerramento manual' estiver habilitada para o gatilho, então é possível encerrar manualmente eventos problema gerados pelo gatilho. Isto é feito no Frontend quando **atualizando um problema**. O evento não é encerrado diretamente - em vez disso uma tarefa 'encerrar evento' é criada, a qual é tratada pelo gerenciador de tarefa. O gerenciador de tarefa gerará um evento OK correspondente e o evento problema será encerrado.

2 Outras fontes de evento

Eventos de serviço

Eventos de serviço são gerados apenas se ações de serviço para estes eventos estão habilitadas. Neste caso, cada alteração de estado de serviço cria um novo evento:

- Evento PROBLEMA - quando o estado do serviço é alterado de OK para PROBLEMA
- Evento OK - quando o estado do serviço é alterado de PROBLEMA para OK

O evento contém detalhes da alteração de estado do serviço - quando ocorreu e qual é o novo estado.

Eventos de descoberta

O Zabbix periodicamente examina os intervalos de IP nas regras de descoberta de rede. A frequência da verificação é configurável para cada regra individualmente. Uma vez que um host ou um serviço é descoberto, um evento de descoberta (ou vários eventos) é gerado.

O Zabbix gera os seguintes eventos:

Evento	Quando gerado
Serviço Up	Sempre que o Zabbix detecta um serviço ativo.
Serviço Down	Sempre que o Zabbix não consegue detectar um serviço.
Host Up	Se ao menos um dos serviços está UP para o IP.
Host Down	Se todos os serviços não estão respondendo.
Serviço Descoberto	Se o serviço retorna após um período de inatividade ou é descoberto pela primeira vez.
Serviço Perdido	Se o serviço é perdido após estar ativo.
Host Descoberto	Se o host retorna após um período de inatividade ou é descoberto pela primeira vez.
Host Perdido	Se o host é perdido após estar ativo.

Eventos de autoregistro de Active Agent

O autoregistro de agentes ativos cria eventos no Zabbix.

Se configurado, um evento de autoregistro de agente ativo é criado quando um agente ativo previamente desconhecido solicita por verificações ou se os metadados do host foram modificados. O Server adiciona um novo host autoregistrado, usando o endereço IP recebido e a porta do agente.

Para mais informações, consulte a página de [autoregistro de agente ativo](#).

Eventos internos

Eventos internos ocorrem quando:

- um item muda de estado 'normal' para 'não suportado'
- um item muda de estado 'não suportado' para 'normal'
- uma regra de descoberta de baixo-nível muda de estado 'normal' para 'não suportado'
- uma regra de descoberta de baixo-nível muda de estado 'não suportado' para 'normal'
- um gatilho muda de estado 'normal' para 'desconhecido'
- um gatilho muda de estado 'desconhecido' para 'normal'

Eventos internos são suportados desde o Zabbix 2.2. O objetivo em introduzir eventos internos é permitir que os usuários sejam notificados quando qualquer evento interno ocorrer, por exemplo, um item se tornar não suportado e parar de reunir dados.

Eventos internos são criados somente quando ações internas para estes eventos estão habilitadas. Para parar a geração de eventos internos (por exemplo, para itens se tornando não suportados), desabilite todas as ações para eventos internos em Configuração → Ações → Ações internas.

Note:

Se ações internas são desabilitadas enquanto um objeto está no estado 'não suportado', um evento de recuperação para este objeto ainda será criado.

Se ações internas são habilitadas enquanto um objeto está no estado 'não suportado', um evento de recuperação para este objeto será criado, mesmo que um 'evento problema' não tenha sido criado para o objeto.

3 Encerramento manual de problemas

Visão geral

Enquanto geralmente os eventos de problema são automaticamente resolvidos quando o estado de um gatilho vai de 'Problema' para 'OK', podem existir casos quando é difícil determinar se um problema foi resolvido por meio de uma expressão de gatilho. Em tais casos, o problema precisa ser resolvido manualmente.

Por exemplo, syslog pode reportar que alguns parâmetros de kernel precisam ser ajustados para melhor performance. Neste caso a questão é reportada aos administradores Linux, eles a corrigem e então encerram o problema manualmente.

Problemas podem ser encerrados manualmente apenas para gatilhos com a opção Permitir encerramento manual habilitada.

Quando um problema é "encerrado manualmente", o Zabbix gera uma nova tarefa interna para o Zabbix server. Então o processo gerenciador de tarefa executa esta tarefa e gera um evento OK, encerrando assim o evento de problema.

Um problema encerrado manualmente não significa que o gatilho subjacente nunca mais entrará em estado de 'Problema' novamente. A expressão do gatilho é reavaliada e pode resultar em um problema:

- Quando novos dados chegam para qualquer item incluído na expressão do gatilho (note que os valores descartados por etapa de pré-processamento de limitação não são considerados como recebidos e não causarão uma reavaliação da expressão do gatilho);

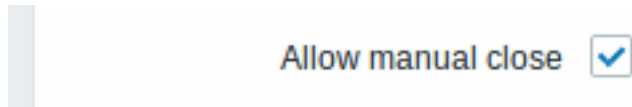
- Quando funções baseadas em tempo são usadas na expressão. Uma lista completa das funções baseadas em tempo pode ser encontrada na [página de gatilhos](#).

Configuração

Duas etapas são necessárias para encerrar um problema manualmente.

Configuração de gatilho

Na configuração do gatilho, habilite a opção Permitir encerramento manual.



Janela atualização de problema

Se um problema surge para um gatilho com a marcação Encerramento manual, você pode abrir a janela [atualização de problema](#) desse problema e encerrar o problema manualmente.

Para encerrar o problema, marque a opção Encerrar problema no formulário e clique em Atualizar.

Update problem

Message

Fixed, closing.

History

Time User User action Message

Scope

☒ Only selected problem
 ☐ Selected and all other problems of related triggers 1 event

Change severity

☐ Not classified
 ☐ Information
 ☐ Warning
 ☐ Average
 ☐ High
 ☐ Disaster

Acknowledge

☐

Close problem

☒

* At least one update operation or message must exist.

Update

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho

A requisição é processada pelo Zabbix Server. Normalmente ele levará alguns poucos segundos para encerrar o problema. Durante esse processo ENCERRANDO é exibido em Monitoramento → Problemas como o estado do problema.

Verificação

Pode ser verificado que um problema foi encerrado manualmente:

- nos detalhes do evento, disponível através de Monitoramento → Problemas;
- pelo uso da macro {EVENT.UPDATE.HISTORY} nas mensagens de notificação que fornecerão esta informação.

5 Correlação de evento

Visão geral

A correlação de evento permite correlacionar eventos problema à sua resolução de uma maneira que é muito precisa e flexível.

A correlação de evento pode ser definida:

- **no nível de gatilho** - um gatilho pode ser usado para relacionar problemas separados à sua solução
- **globalmente** - problemas podem ser correlacionados à sua solução a partir de um gatilho/método polling diferente usando regras de correlação globais

1 Correlação de evento baseada em gatilho

Visão geral

A correlação de evento baseada em gatilho permite correlacionar problemas separados reportados por um gatilho.

Enquanto de forma geral um evento OK pode encerrar todos os eventos problema criados por um gatilho, há casos onde uma abordagem mais detalhada é necessária. Por exemplo, quando monitorando arquivos de log você pode querer descobrir certos problemas em um arquivo de log e encerrá-los individualmente em vez de todos ao mesmo tempo.

Este é o caso com gatilhos que possuem habilitada a Geração de Eventos Problema Múltiplos. Tais gatilhos são normalmente usados para monitoramento de log, processamento de trap, etc.

No Zabbix é possível relacionar eventos problema baseado em **etiquetagem**. Etiquetas são usadas para extrair valores e criar identificação para eventos problema. Tirando vantagem disto, problemas podem também ser encerrados individualmente com base na correspondência de etiqueta.

Em outras palavras, o mesmo gatilho pode criar eventos separados identificados pela etiqueta do evento. Assim eventos problema podem ser identificados um a um e encerrados separadamente com base na identificação pela etiqueta do evento.

Como isso funciona

No monitoramento de log você pode encontrar linhas similares a estas:

```
Linha1: Aplicação 1 parada
Linha2: Aplicação 2 parada
Linha3: Aplicação 1 foi reiniciada
Linha4: Aplicação 2 foi reiniciada
```

A ideia da correlação de evento é estar apto a corresponder o evento problema da Linha1 à resolução da Linha3 e o evento problema da Linha2 à resolução da Linha4, e encerrar estes problemas um a um:

```
Linha1: Aplicação 1 parada
Linha3: Aplicação 1 foi reiniciada #problema da Linha 1 encerrado
```

```
Linha2: Aplicação 2 parada
Linha4: Aplicação 2 foi reiniciada #problema da Linha 2 encerrado
```

Para fazer isto você precisa etiquetar estes eventos relacionados como, por exemplo, "Aplicação 1" e "Aplicação 2". Isto pode ser feito pela aplicação de uma expressão regular à linha de log para extrair o valor da etiqueta. Então, quando eventos são criados, eles são etiquetados como "Aplicação 1" e "Aplicação 2" respectivamente e o problema pode ser correspondido à resolução.

Configuração

Item

Para começar, você pode querer definir um item que monitora um arquivo de log, por exemplo:

```
log[/var/log/syslog]
```

Item	Tags	Preprocessing
		<div><div>* Name</div><div>Syslog</div></div> <div><div>Type</div><div>Zabbix agent (active) ▾</div></div> <div><div>* Key</div><div>log[/var/log/syslog]</div></div> <div><div>Type of information</div><div>Text ▾</div></div> <div><div>* Update interval</div><div>30s</div></div>

Com o item configurado, aguarde um minuto para que as mudanças de configuração sejam capturadas e então vá até **últimos dados** para certificar-se de que o item iniciou a coleta de dados.

Gatilho

Com o item funcionando você precisa configurar o **gatilho**. É importante decidir quais entradas no arquivo de log são dignas de atenção. Por exemplo, a seguinte expressão de gatilho buscará por uma string como 'Parando (Stopping)' para sinalizar possíveis problemas:

```
find(/My host/log[/var/log/syslog],,"regexp","Parando")=1
```

Attention:

Para certificar-se que cada linha contendo uma string "Parando" é considerada um problema também defina o Modo geração de evento problema na configuração do gatilho para 'Múltiplo'.

Então defina uma expressão de recuperação. A seguinte expressão de recuperação solucionará todos os problemas se uma linha de log é encontrada contendo a string "Iniciando (Starting)":

```
find(/My host/log[/var/log/syslog],,"regexp","Iniciando")=1
```

Como nós não queremos este comportamento, é importante certificar-se de alguma forma de que os problemas raiz sejam encerrados, e não todos os problemas. É aqui onde a etiquetagem pode ajudar.

Problemas e soluções podem ser correspondidos especificando uma etiqueta na configuração do gatilho. As seguintes configurações devem ser feitas:

- Modo geração de evento problema: Múltiplo
- Evento OK encerra: Todos os problemas se valores de etiqueta corresponderem
- Informe o nome da etiqueta para correspondência de evento

Trigger
Tags
Dependencies

* Name
Service {{ITEM.VALUE}.regsub("^.* service ([a-zA-Z]*) .*\$", "\1")} stopped

Event name
Service {{ITEM.VALUE}.regsub("^.* service ([a-zA-Z]*) .*\$", "\1")} stopped

Operational data

Severity
Not classified
Information
Warning
Average
High
Disas

* Problem expression

```
find(/My host/log[/var
/log/syslog],,"regex","Stopping")=1
```

Add

[Expression constructor](#)

OK event generation
Expression
Recovery expression
None

* Recovery expression

```
find(/My host/log[/var
/log/syslog],,"regex","Starting")=1
```

Add

[Expression constructor](#)

PROBLEM event generation mode
Single
Multiple

OK event closes
All problems
All problems if tag values match

* Tag for matching
Service

- configure as **etiquetas** para extrair os valores de etiqueta das linhas do log

Trigger
Tags 2
Dependencies

Trigger tags
Inherited and trigger tags

Name
Value

Datacenter
value

Service
{{ITEM.VALUE}.regsub("^.* service ([a-zA-Z]*) .*\$", "\1")}

Add

Se configurado com sucesso você conseguirá ver os eventos problema etiquetados por aplicação e correspondidos à sua solução em Monitoramento → Problemas.

Problems
Export to CSV

Time	Severity	Recovery time	Status	Info	Host	Problem	Duration	Ack	Actions	Tags
15:28:13	High	15:28:25	RESOLVED	Zabbix server	Service Apache stopped	12s	No		Service: Apache	Webserver

Warning:

Devido possibilidade de má configuração, quando etiquetas de evento similares podem ser criadas para problemas **não relacionados**, por favor revise os casos destacados abaixo!

- Com duas aplicações gravando mensagens de erro e recuperação no mesmo arquivo de log um usuário deve decidir usar duas etiquetas Aplicação no mesmo gatilho com valores de etiqueta diferentes pelo uso de expressões regulares separadas nos valores de etiqueta para extrair os nomes de, digamos, aplicação A e aplicação B da macro {ITEM.VALUE} (p.e. quando os formatos de mensagem diferem). No entanto, isto pode não funcionar como planejado se não houver correspondência com as expressões regulares. Expressões regulares não correspondidas renderão valores de etiqueta vazios e um único valor de etiqueta vazio em ambos os eventos PROBLEMA e OK é suficiente para correlacioná-los. Então uma mensagem de recuperação da aplicação A podem acidentalmente encerrar uma mensagem de erro da aplicação B.
- Etiquetas vigentes e valores de etiqueta só se tornam visíveis quando um gatilho é disparado. Se a expressão regular utilizada for inválida, ela é silenciosamente substituída por uma string *DESCONHECIDO*. Se o evento problema inicial com um valor de etiqueta *DESCONHECIDO* é ignorado, podem existir eventos OK subsequentes com o mesmo valor de etiqueta *DESCONHECIDO* que podem encerrar eventos problema que não deveriam ter sido encerrados.
- Se um usuário usa a macro {ITEM.VALUE} sem funções de macro como valor de etiqueta, a limitação de 255 caracteres se aplica. Quando mensagens de log são longas e os primeiros 255 caracteres não são específicos, isto pode também resultar em etiquetas de evento similares para problemas não relacionados.

2 Correlação de eventos globais

Visão geral

A correlação global de eventos permite alcançar todas as métricas monitoradas pelo Zabbix e criar correlações.

É possível correlacionar eventos criados por triggers e aplicar as mesmas operações a todos eles. Ao criar regras de correlação inteligentes é realmente possível salvar a si mesmo de milhares de notificações repetitivas e foco nas causas raiz de um problema!

A correlação de eventos globais é um mecanismo poderoso, que permite desamarre-se do problema baseado em um trigger e da lógica de resolução. Assim até agora, um único evento de problema foi criado por um trigger e for dependente desse mesmo trigger para a resolução do problema. Nós não pudemos resolver um problema criado por um trigger com outro trigger. Mas com correlação de eventos com base na marcação de eventos, podemos.

Por exemplo, um log de trigger pode relatar problemas de aplicativo, enquanto um o trigger de pesquisa pode relatar que o aplicativo está funcionando. Tirando vantagem das tags de evento, você pode marcar o trigger de log como Status: Down enquanto marca o trigger de checagem como Status: Up. Então, em um mundo regra de correlação, você pode relacionar esses triggers e atribuir um operação a essa correlação, como fechar os eventos antigos.

Em outro uso, a correlação global pode identificar triggers semelhantes e aplicar a mesma operação a eles. E se pudéssemos obter apenas um problema relatório por problema de porta de rede? Não há necessidade de denunciá-los todos. Aquilo é também é possível com a correlação de eventos globais.

A correlação de eventos globais é configurada em **regras de correlação**. Uma regra de correlação define como os novos eventos de problema são emparelhados com eventos de problemas existentes e o que fazer em caso de correspondência (fecha o novo evento, fecha eventos antigos correspondentes gerando eventos correspondentes como OK). Se um problema for fechado por correlação global, ele será relatado no Coluna de Info em Monitoramento → Problemas.

A configuração de regras de correlação global está disponível apenas para usuários de nível super administrador.

Attention:

A correlação de eventos deve ser configurada muito com cuidado, pois pode afetar negativamente o desempenho do processamento de eventos ou, se configurado incorretamente, feche mais eventos do que o pretendido (no pior caso até mesmo todos os eventos de problema podem ser fechados).

Para configurar a correlação global **com segurança**, observe o seguinte dicas importantes:

- Reduza o escopo de correlação. Sempre defina uma tag exclusiva para o novo evento que está emparelhado com eventos antigos e use a Nova tag de evento condição de correlação;
- Adicione uma condição com base no evento antigo ao usar o Operação de fechar evento antigo (ou então todos os problemas existentes podem ser fechados);
- Evite usar nomes de tags comuns que podem acabar sendo usados por diferentes configurações de correlação;
- Mantenha o número de regras de correlação limitadas àquelas que você realmente necessidade.

Veja também: [questões frequentes](#).

Configuration

To configure event correlation rules globally:

- Go to Configuration → Event correlation
- Click on Create correlation to the right (or on the correlation name to edit an existing rule)
- Enter parameters of the correlation rule in the form

* Name

Close old events

Type of calculation

And

▼

A and (B and C) and D

* Conditions

Label	Name
A	Value of old event tag <i>Application</i> equals value of new event tag <i>Application</i>
B	Value of old event tag <i>Application</i> equals <i>ABC</i>
C	Value of old event tag <i>State</i> equals <i>Down</i>
D	Value of new event tag <i>State</i> equals <i>Up</i>
Add	

Description

Close old events for application ABC if an event with State=Up happens.

Operations

☒ Close old events
☐ Close new event

* At least one operation must be selected.

Enabled

☒

All mandatory input fields are marked with a red asterisk.

Parameter	Description
Name	Unique correlation rule name.
Type of calculation	<p>The following options of calculating conditions are available:</p> <p>And - all conditions must be met</p> <p>Or - enough if one condition is met</p> <p>And/Or - AND with different condition types and OR with the same condition type</p> <p>Custom expression - a user-defined calculation formula for evaluating action conditions. It must include all conditions (represented as uppercase letters A, B, C, ...) and may include spaces, tabs, brackets (), and (case sensitive), or (case sensitive), not (case sensitive).</p>
Conditions	List of conditions. See below for details on configuring a condition.
Description	Correlation rule description.
Operations	<p>Mark the checkbox of the operation to perform when event is correlated. The following operations are available:</p> <p>Close old events - close old events when a new event happens. Always add a condition based on the old event when using the Close old events operation or all existing problems could be closed.</p> <p>Close new event - close the new event when it happens</p>

Parameter	Description
Enabled	If you mark this checkbox, the correlation rule will be enabled.

To configure details of a new condition, click on [Add](#) in the Conditions block. A popup window will open where you can edit the condition details.

New condition

Type

New event tag value

Tag

State

Operator

equals

does not equal

contains

does not contain

Value

Up

Add

Cancel

Parameter	Description
New condition	<p>Select a condition for correlating events.</p> <p>Note that if no old event condition is specified, all old events may be matched and closed. Similarly if no new event condition is specified, all new events may be matched and closed.</p> <p>The following conditions are available:</p> <p>Old event tag - specify the old event tag for matching.</p> <p>New event tag - specify the new event tag for matching.</p> <p>New event host group - specify the new event host group for matching.</p> <p>Event tag pair - specify new event tag and old event tag for matching. In this case there will be a match if the values of the tags in both events match. Tag names need not match. This option is useful for matching runtime values, which may not be known at the time of configuration (see also Example 1).</p> <p>Old event tag value - specify the old event tag name and value for matching, using the following operators:</p> <p>equals - has the old event tag value</p> <p>does not equal - does not have the old event tag value</p> <p>contains - has the string in the old event tag value</p> <p>does not contain - does not have the string in the old event tag value</p> <p>New event tag value - specify the new event tag name and value for matching, using the following operators:</p> <p>equals - has the new event tag value</p> <p>does not equal - does not have the new event tag value</p> <p>contains - has the string in the new event tag value</p> <p>does not contain - does not have the string in the new event tag value</p>

Warning:

Because misconfiguration is possible, when similar event tags may be created for **unrelated** problems, please review the cases outlined below!

- Actual tags and tag values only become visible when a trigger fires. If the regular expression used is invalid, it is silently replaced with an *UNKNOWN* string. If the initial problem event with an *UNKNOWN* tag value is missed, there may appear subsequent OK events with the same *UNKNOWN* tag value that may close problem events which they shouldn't have closed.
- If a user uses the {ITEM.VALUE} macro without macro functions as the tag value, the 255-character limitation applies. When log messages are long and the first 255 characters are non-specific, this may also result in similar event tags for unrelated problems.

Exemplo

Interrompa eventos problemáticos repetitivos a partir da mesma porta de rede.

* Name	Correlate network port problems	
Type of calculation	And	A and B
* Conditions	Label	Name
	A	Value of old event tag <i>Port</i> equals value of new event tag <i>Port</i>
	B	Value of old event tag <i>Host</i> equals value of new event tag <i>Host</i>
	Add	
Description	Keep only one problem per port. No need to report all of them.	
Operations	<input type="checkbox"/> Close old events <input checked="" type="checkbox"/> Close new event	
	* At least one operation must be selected.	
Enabled	<input checked="" type="checkbox"/>	
	Add	Cancel

Essa regra de correlação global irá correlacionar problemas se os valores de tag * Host* e a porta existirem no trigger e forem iguais no evento original e no novo evento.

Essa operação encerrará eventos problemáticos novos na mesma porta de rede, mantendo somente o problema original aberto.

6 Marcação

Visão geral

Existe uma opção para marcar várias entidades no Zabbix. As etiquetas podem ser definido para:

- modelos
- anfitriões
- Itens
- cenários da web

- gatilhos
- itens de modelo e gatilhos
- protótipos de host, item e gatilho

As tags têm vários usos, principalmente para marcar eventos. Se as entidades são marcadas, os novos eventos correspondentes são marcados de acordo:

- com modelos marcados - quaisquer problemas de host criados por entidades (itens, gatilhos, etc) deste modelo serão marcadas
- com hosts marcados - qualquer problema do host será marcado
- com itens marcados, cenários da web - qualquer dado/problema deste item ou cenário da web será marcado
- com gatilhos marcados - qualquer problema deste gatilho será marcado

Um evento de problema herda todas as tags de toda a cadeia de modelos, hosts, itens, cenários da web, gatilhos. tag:value completamente idêntico combinações (após macros resolvidas) são mescladas em uma em vez de sendo duplicado, ao marcar o evento.

Ter tags de evento personalizadas permite mais flexibilidade. Importante, eventos podem ser **correlacionados** com base em etiquetas de eventos. Em outros usos, as ações podem ser definidas com base em tags eventos. Os problemas de itens podem ser agrupados com base em tags.

A marcação é realizada como um par de tag name e value. Você pode usar apenas o nome ou emparelhá-lo com um valor:

MySQL, Serviço:MySQL, Serviços, Serviços:Cliente, Aplicativos, Aplicativo:Java, Prioridade:Alta

Uma entidade (modelo, host, item, cenário da web, gatilho ou evento) pode ser marcadas com o mesmo nome, mas com valores diferentes - essas tags não serão considerados 'duplicados'. Da mesma forma, uma tag sem valor e a mesma tag com valor pode ser usado simultaneamente.

Casos de uso

Alguns casos de uso para esta funcionalidade são os seguintes:

1. Marcar eventos de gatilho no frontend
 - Definir tags em nível de trigger;
 - Veja como todos os problemas de gatilho são marcados com essas tags em Monitoramento → Problemas.
2. Marque todos os problemas herdados do modelo
 - Definir uma tag no nível do template, por exemplo 'App=MySQL';
 - Veja como esses problemas de host que são criados por gatilhos de este modelo está marcado com essas tags em Monitoramento → Problemas.
3. Marque todos os problemas do host
 - Definir uma tag no nível do host, por exemplo 'Service=JIRA';
 - Veja como todos os problemas dos gatilhos do host estão marcados com estes tags em Monitoramento → Problemas
4. Agrupar itens relacionados
 - Definir uma tag no nível do item, por exemplo 'MySQL';
 - Veja todos os itens marcados como 'MySQL' em Últimos dados usando o filtro de tags
5. Identifique problemas em um arquivo de log e feche-os separadamente
 - Definir tags no gatilho de log que identificarão eventos usando extração de valor pela macro `{{ITEM.VALUE<N>}}.regsub()`;
 - Na configuração do acionador, tenha geração de vários eventos de problema modo;
 - Na configuração do acionador, use **evento correlação**: selecione o opção que o evento OK fecha apenas os eventos correspondentes e escolha a opção tag para correspondência;
 - Veja os eventos de problema criados com uma tag e fechados individualmente.
6. Use-o para filtrar notificações
 - Definir tags no nível do gatilho para marcar eventos por diferentes Tag;
 - Use a filtragem de tags em condições de ação para receber notificações apenas nos eventos que correspondem aos dados da tag.
7. Use as informações extraídas do valor do item como valor da etiqueta
 - Use uma macro `{{ITEM.VALUE<N>}}.regsub()` no valor da tag;
 - Veja os valores das tags em Monitoramento → Problemas como dados extraídos do valor do item.
8. Identifique melhor os problemas nas notificações
 - Definir tags no nível de trigger;
 - Use uma macro `{EVENT.TAGS}` na notificação do problema;
 - Mais fácil identificar qual aplicativo/serviço a notificação pertence a.
9. Simplifique as tarefas de configuração usando tags no nível do modelo
 - Definir tags no nível de trigger do template;
 - Veja essas tags em todos os acionadores criados a partir de acionadores de modelo.
10. Crie acionadores com tags da descoberta de baixo nível (LLD)
 - Definir tags em protótipos de trigger;

- Use macros LLD no nome ou valor da tag;
- Veja essas tags em todos os gatilhos criados a partir de protótipos de gatilho.

Configuração

As tags podem ser inseridas em uma guia dedicada, por exemplo, no gatilho configuração:

Name	Value	Action
Cloud	value	Remove
Service	MySQL	Remove
Customers	value	Remove
Host	{{ITEM.VALUE2}.iregsub(pattern, output)}	Remove

[Add](#)

Suporte a macros

As seguintes macros podem ser usadas em tags de acionador:

- {ITEM.VALUE}, {ITEM.LASTVALUE}, {HOST.HOST}, {HOST.NAME}, Macros {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {HOST.PORT} e {HOST.ID} pode ser usado para preencher o nome da tag ou o valor da tag
- {INVENTÁRIO.*} **macros** podem ser usados para referenciar valores de inventário de host de um ou vários hosts em um expressão de gatilho
- **Macros do usuário** e macro do usuário context é compatível com o nome/valor da tag. O contexto da macro do usuário pode incluir macros de descoberta de baixo nível
- Macros de descoberta de baixo nível podem ser usadas para o nome/valor da tag em acionar protótipos

As seguintes macros podem ser usadas em notificações baseadas em gatilho:

- As macros {EVENT.TAGS} e {EVENT.RECOVERY.TAGS} serão resolvidas para um lista separada por vírgulas de tags de evento ou tags de evento de recuperação
- As macros {EVENT.TAGSJSON} e {EVENT.RECOVERY.TAGSJSON} serão resolvidas para uma matriz JSON contendo tag de evento **objects** ou recuperação objetos de tag de evento

As seguintes macros podem ser usadas em template, host, item e web tags de cenário:

- {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, Macros {HOST.PORT} e {HOST.ID}
- {INVENTÁRIO.*} **macros**
- **Macros do usuário**
- Macros de descoberta de baixo nível podem ser usadas em tags de protótipo de item

As seguintes macros podem ser usadas em tags de protótipo de host:

- {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, Macros {HOST.PORT} e {HOST.ID}
- {INVENTÁRIO.*} **macros**
- **Macros do usuário**
- **Macros de descoberta de baixo nível** ser resolvido durante o processo de descoberta e, em seguida, adicionado ao host descoberto

Extração de substring em tags de gatilho

A extração de substring é suportada para preencher o nome da tag ou a tag valor, usando uma macro **função** - aplicando uma expressão regular ao valor obtido por {ITEM.VALUE}, Macro {ITEM.LASTVALUE} ou uma macro de descoberta de baixo nível. Por exemplo:

```
{{ITEM.VALUE}.regsub(padrao, saida)}
{{ITEM.VALUE}.iregsub(padrao, saida)}
```

```
{{#LLDMACRO}}.regsub(padrao, saida)}  
{{#LLDMACRO}}.iregsub(padrao, saida)}
```

O nome e o valor da tag serão reduzidos para 255 caracteres se o comprimento exceder 255 caracteres após a resolução macro.

Consulte também: Usando funções de macro em [descoberta de baixo nível macros](#) para marcação de eventos.

Visualizando tags de evento

A marcação, se definida, pode ser vista com novos eventos em:

- Monitoramento → Problemas
- Monitoramento → Problemas → Detalhes do evento
- Monitoramento → Painel → Problemas widget (na janela pop-up que abre ao passar o mouse sobre o nome do problema)

Status	Info	Host	Problem	Duration	Ack	Actions	Tags
PROBLEM		New host	Nodata on 'New host' for two minutes	39s	No		Cloud Customers Host: HP-Pro

Cloud Customers Host: HP-Pro Service: MySQL

Apenas as três primeiras entradas de tag são exibidas. Se houver mais de três entradas de tag, é indicado por três pontos. Se você rolar o mouse sobre esses três pontos, todas as entradas de tags são exibidas em uma janela pop-up.

Observe que a ordem em que as tags são exibidas é afetada pela tag filtragem e a opção Prioridade de exibição de tags no filtro de Monitoramento → Problemas ou o widget do painel Problemas.

7 Visualização

1 Gráficos

Visão geral

Com muitos dados fluindo para o Zabbix, fica muito mais fácil para o usuários se eles puderem ver uma representação visual do que está acontecendo em vez de apenas números.

É aqui que entram os gráficos. Os gráficos permitem apreender o fluxo de dados em um olhar, correlacionar problemas, descobrir quando algo começou ou fazer uma apresentação de quando algo pode se tornar um problema.

O Zabbix fornece aos usuários:

- [gráficos simples] integrados (/manual/config/visualization/graphs/simple) de dados de um item
- a possibilidade de criar mais complexos [personalizados gráficos](#)
- acesso a uma comparação de vários itens rapidamente em [ad-hoc gráficos](#)
- moderno personalizável [vetor gráficos](#)

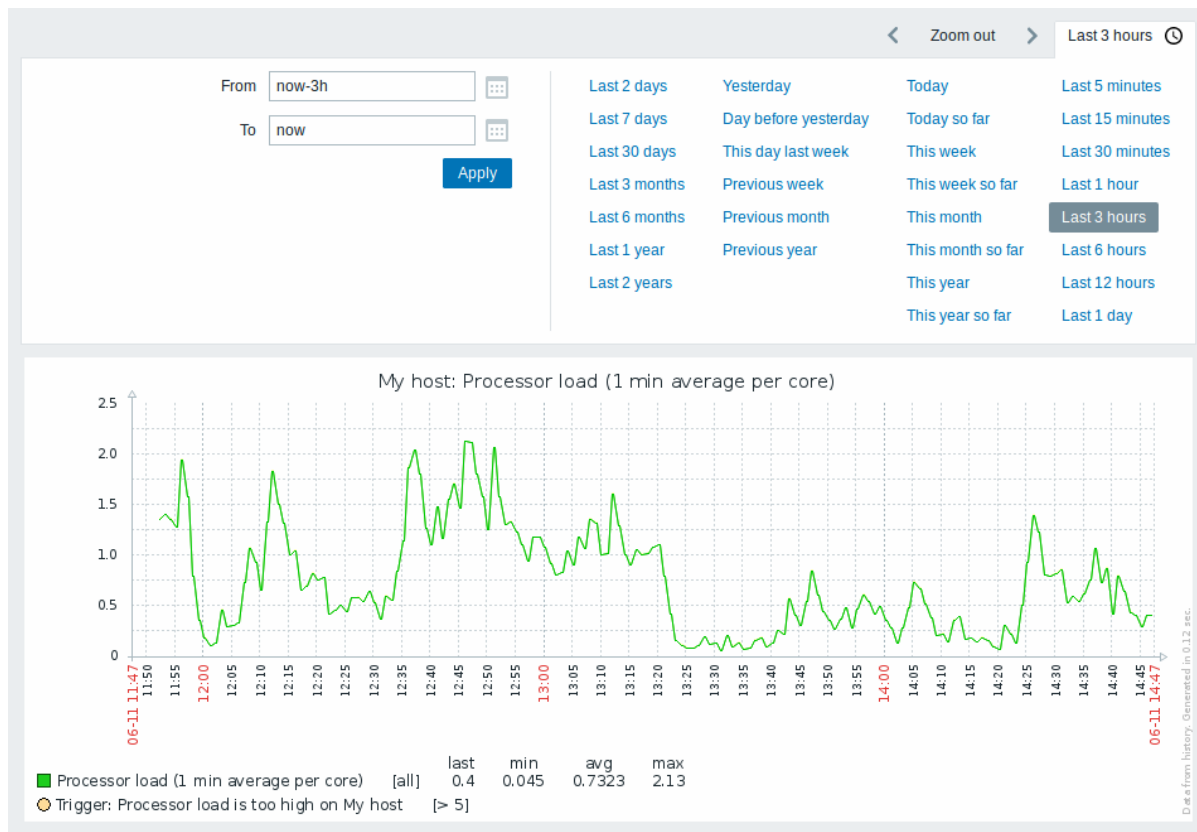
1 Gráficos simples

Visão geral

Gráficos simples são fornecidos para a visualização de dados coletados por Itens.

Nenhum esforço de configuração é necessário por parte do usuário para visualizar gráficos. Eles são disponibilizados gratuitamente pelo Zabbix.

Basta ir para Monitoring → Últimos dados e clicar no link Gráfico para o respectivo item e um gráfico serão exibidos.




Note:

Gráficos simples são fornecidos para todos os itens numéricos. Para itens textuais, um link para Histórico está disponível em Monitoramento → Últimas dados.

Seletor de período de tempo

Anote o seletor de período de tempo acima do gráfico. Ele permite selecionar períodos frequentemente necessários com um clique do mouse.

Observe que opções como Hoje, Esta semana, Este mês, Este ano exibem todo o período, incluindo as horas/dias no futuro. Hoje até agora, ao contrário, mostra apenas as horas passadas.

Uma vez que um período é selecionado, ele pode ser movido para frente e para trás no tempo clicando no  botões de seta. O botão Zoom out permite diminuir o zoom do período dois vezes ou em 50% em cada direção. Diminuir o zoom também é possível por clicando duas vezes nos gráficos. O seletor de período de tempo inteiro pode ser recolhido clicando no rótulo da guia que contém o período selecionado corda.

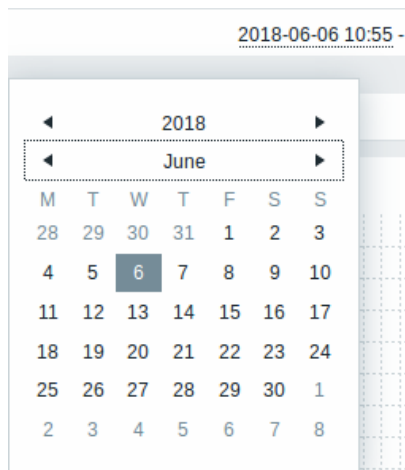
Os campos De/Para exibem o período selecionado em:

- sintaxe de tempo absoluto no formato Y-m-d H:i:s
- sintaxe de tempo relativo, por exemplo: now-1d

Uma data em formato relativo pode conter uma ou várias operações (- ou +), e. agora-1d ou agora-1d-2h+5m. Por tempo relativo as seguintes abreviações são suportadas:

- agora
- s (segundos)
- m (minutos)
- h (horas)
- d (dias)
- w (semanas)
- M (meses)
- y (anos)

É possível escolher uma data de início/término específica clicando no botão ícone de calendário ao lado dos campos De/Para. Neste caso, a data pop-up do seletor será aberto.



Dentro do selecionador de datas, é possível navegar entre os blocos de ano/mês/data usando Tab e Shift+Tab. Setas ou setas do teclado botões permitem selecionar o valor desejado. Pressionando Enter (ou clicando no valor desejado) ativa a escolha.

Outra maneira de controlar o tempo exibido é destacar uma área na o gráfico com o botão esquerdo do mouse. O gráfico ampliará o área destacada assim que você soltar o botão esquerdo do mouse.

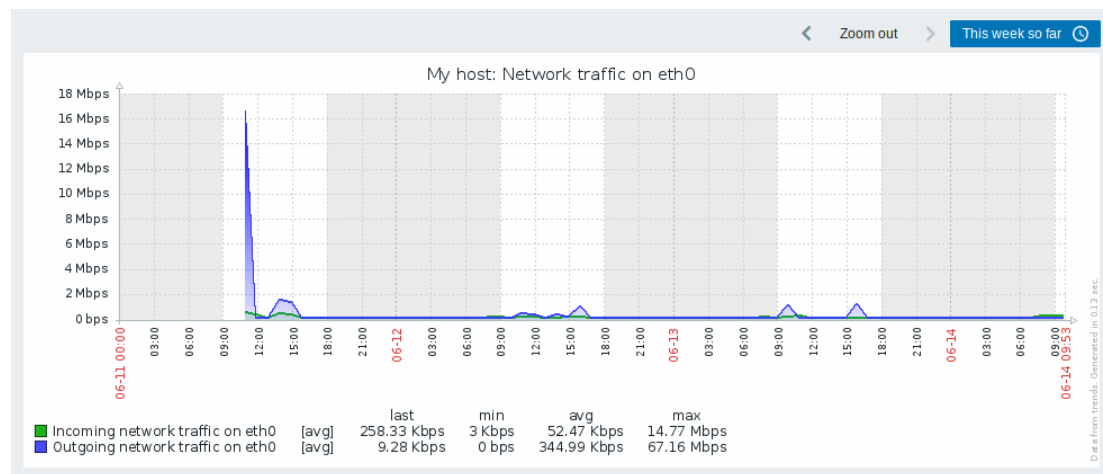
Caso nenhum valor de tempo seja especificado ou o campo seja deixado em branco, o valor de tempo será definido como "00:00:00". Isso não se aplica à seleção de data de hoje: nesse caso o tempo será ajustado para o valor atual.

Dados recentes vs períodos mais longos

Para dados muito recentes, uma linha **única** é desenhada conectando cada valor. A linha única é desenhada enquanto houver pelo menos uma pixel horizontal disponível para um valor.

Para dados que mostram um período mais longo, **três linhas** são desenhadas - um escuro verde mostra a média, enquanto uma linha rosa claro e uma linha verde clara mostra os valores máximo e mínimo naquele momento. O espaço entre os altos e os baixos é preenchido com fundo amarelo.

O tempo de trabalho (dias úteis) é exibido em gráficos como um branco fundo, enquanto o tempo de folga é exibido em cinza (com o Azul original tema frontend padrão).



O tempo de trabalho é sempre exibido em gráficos simples, enquanto exibi-lo em **gráficos personalizados** é um usuário preferência.

O tempo de trabalho não é exibido se o gráfico mostrar mais de 3 meses.

Linhas de gatilho

Gatilhos simples são exibidos como linhas com traços pretos sobre o gatilho cor de gravidade -- observe a linha azul no gráfico e a informações de disparo exibidas na legenda. Até 3 linhas de gatilho podem ser exibido no gráfico; se houver mais gatilhos, então os gatilhos com menor gravidade são priorizados. Os gatilhos são sempre exibidos em gráficos simples, enquanto exibi-los em **custom gráficos** é uma preferência do usuário.



Gerando a partir do histórico/tendências

Os gráficos podem ser desenhados com base em qualquer item **histórico** ou **tendências**.

Para os usuários que têm frontend **debug mode** ativado, um cinza, vertical legenda é exibida no canto inferior direito de um gráfico indicando onde o dados vêm.

Vários fatores influenciam se o histórico de tendências é usado:

- longevidade do histórico de itens. Por exemplo, o histórico do item pode ser mantido para 14 dias. Nesse caso, quaisquer dados com mais de quatorze dias serão vindo de tendências.
- congestionamento de dados no gráfico. Se a quantidade de segundos a ser exibida em um pixel gráfico horizontal excede 3600/16, os dados de tendência são exibidos (mesmo que o histórico do item ainda esteja disponível para o mesmo período).
- se as tendências estiverem desabilitadas, o histórico do item é usado para a construção do gráfico - se disponível para esse período. Isso é suportado a partir do Zabbix 2.2.1 (antes, tendências desabilitadas significavam um gráfico vazio para o mesmo que o histórico do item estivesse disponível).

Ausência de dados

Para itens com um intervalo de atualização regular, nada é exibido no gráfico se os dados do item não forem coletados.

No entanto, para itens de caçadores e itens com um intervalo de atualização programado (e intervalo de atualização regular definido como 0), uma linha reta é desenhada à esquerda até o primeiro valor arrecadado e do último valor arrecadado até o final do gráfico; a linha está no nível do primeiro/último valor respectivamente.

Mudando para valores brutos

Uma lista suspensa no canto superior direito permite alternar do gráfico simples para as listagens de Valores/500 valores mais recentes. Isso pode ser útil para visualizar os valores numéricos que compõem o gráfico.

Os valores aqui representados são brutos, ou seja, sem unidades ou pós-processamento de valores é usado. O mapeamento de valor, no entanto, é aplicado.

Problemas conhecidos

Consulte **problemas conhecidos** para obter gráficos.

2 Gráficos personalizados

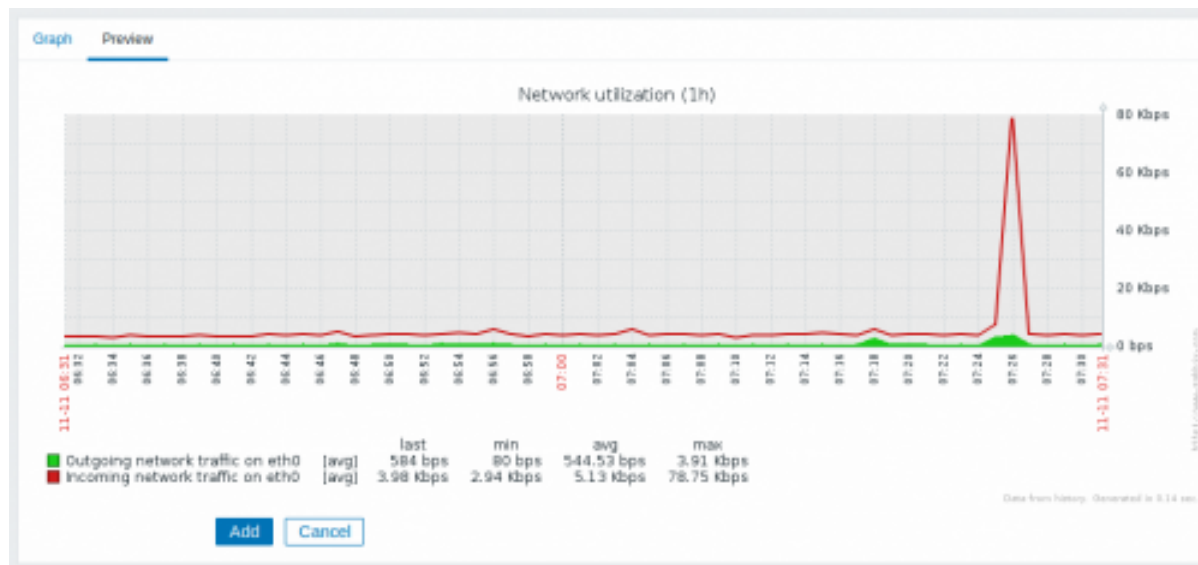
Visão geral

Parâmetro	Descrição
Tipo de gráfico	<p>Tipo de gráfico:</p> <p>Normal - gráfico normal, valores exibidos como linhas</p> <p>Empilhado - gráfico empilhado, áreas preenchidas exibidas</p> <p>Pie - gráfico de pizza</p> <p>Explodido - gráfico de pizza "explodido", partes exibidas como "recortadas" da pizza</p>
Mostrar legenda	Marcar esta caixa para exibir a legenda do gráfico.
Mostrar horário de trabalho	Se selecionado, o horário de folga será mostrado com um fundo cinza. Não disponível para gráficos de pizza e explodidos.
Mostrar acionadores	Se selecionado, os acionadores simples serão exibidos como linhas com traços pretos sobre a cor da gravidade do acionador. Não disponível para gráficos de pizza e explodidos.
Linha de percentil (esquerda)	Exibe o percentil para o eixo Y esquerdo. Se, por exemplo, o percentil 95% for definido, a linha do percentil estará no nível em que 95% dos valores se enquadram. Exibido como uma linha verde brilhante. Disponível apenas para gráficos normais.
Linha do percentil (direita)	Exibe o percentil do eixo Y direito. Se, por exemplo, o percentil 95% for definido, a linha do percentil estará no nível em que 95% dos valores se enquadram. Exibido como uma linha vermelha brilhante. Disponível apenas para gráficos normais.
Valor MIN do eixo Y	<p>Valor mínimo do eixo Y:</p> <p>Calculado - O valor mínimo do eixo Y será calculado automaticamente</p> <p>Fixed - valor mínimo fixo para o eixo Y . Não disponível para gráficos de pizza e pizza explodida.</p>
Valor MAX do eixo Y	<p>Item - o último valor do item selecionado será o valor mínimo</p> <p>Valor máximo do eixo Y:</p> <p>Calculado - o valor máximo do eixo Y será calculado automaticamente</p> <p>Fixed - valor máximo fixo para o eixo Y . Não disponível para gráficos de pizza e pizza explodida.</p>
Visão 3D	Item - o último valor do item selecionado será o valor máximo
Itens	<p>Ativar estilo 3D. Apenas para gráficos de pizza e explodidos.</p> <p>Itens, cujos dados devem ser exibidos neste gráfico. Clique em Adicionar para selecionar itens. Você também pode selecionar várias opções de exibição (função, estilo de desenho, exibição do eixo esquerdo/direito, cor).</p>
Ordem de ordenação	Ordem de saque. O será processado primeiro. Pode ser usado para desenhar linhas ou regiões atrás (ou na frente) de outra. Você pode arrastar e soltar itens pela seta no início de uma linha para definir a ordem de classificação ou qual item é (0→100) exibido na frente do outro .
Nome	O nome do item selecionado é exibido como um link. Clicar no link abre a lista de outros itens disponíveis.
Tipo	<p>Tipo (disponível apenas para gráficos de pizza e pizza explodida):</p> <p>Simple - o valor do item é representado proporcionalmente na pizza</p> <p>Soma do gráfico - o valor o valor do item representa a pizza inteira</p> <p>Observe que a coloração do item "soma do gráfico" só será visível na medida em que não for ocupado por itens "proporcionais".</p>

Parâmetro	Descrição
Função	<p>Selecione quais valores serão exibidos quando houver mais de um valor por pixel do gráfico vertical para um item:</p> <p>all - exibe todos os valores possíveis (mínimo, máximo, médio) no gráfico . Observe que para períodos mais curtos essa configuração não tem efeito; somente por períodos mais longos, quando o congestionamento de dados em um pixel de gráfico vertical aumenta, 'all' começa a exibir valores mínimos, máximos e médios. Esta função só está disponível para o tipo de gráfico Normal. Consulte também: Gerando gráficos do histórico/tendências.</p> <p>avg - exibe os valores médios</p> <p>last - exibe o últimos valores. Esta função só está disponível se Pie/Pie explodida estiver selecionado como tipo de gráfico.</p> <p>max - exibe os valores máximos</p> <p>min - exibe os valores mínimos</p>
Estilo	<p>Selecione o estilo de desenho (disponível apenas para de gráficos normais; para gráficos empilhados, a região de- preenchida é sempre usada) para aplicar aos dados do item -</p> <p>Linha, Linha em negrito, Região preenchida, * Ponto, Linha tracejada, Linha de gradiente*.</p>
Lado do eixo Y	<p>Selecione o lado do eixo Y para mostrar os dados do item -</p> <p>Esquerda, Direita.</p>
Cor	<p>Selecione a cor a ser aplicada aos dados do item.</p>

Visualização do gráfico

Na guia Visualizar, uma visualização do gráfico é exibida para que você possa ver imediatamente o que você está criando.



Observe que a visualização não mostrará nenhum dado para itens de modelo.



Neste exemplo, preste atenção à linha tracejada em negrito exibindo o nível de disparo e as informações de disparo exibidas na legenda.

Note:

Não podem ser exibidas mais de 3 linhas de disparo. Se houver mais gatilhos, então os gatilhos com menor gravidade são priorizados para tela.

Se a altura do gráfico for inferior a 120 pixels, nenhum acionador será exibido na legenda.

3 Gráficos ad hoc

Visão geral

Enquanto um **gráfico simples** é ótimo para acessar dados de um item e **gráficos personalizados** oferecem opções de personalização, nenhum dos dois permitem criar rapidamente um gráfico de comparação para vários itens com pouco esforço e nenhuma manutenção.

Para resolver este problema, desde o Zabbix 2.4 é possível criar ad-hoc gráficos para vários itens de uma forma muito rápida.

Configuração

Para criar um gráfico ad hoc, faça o seguinte:

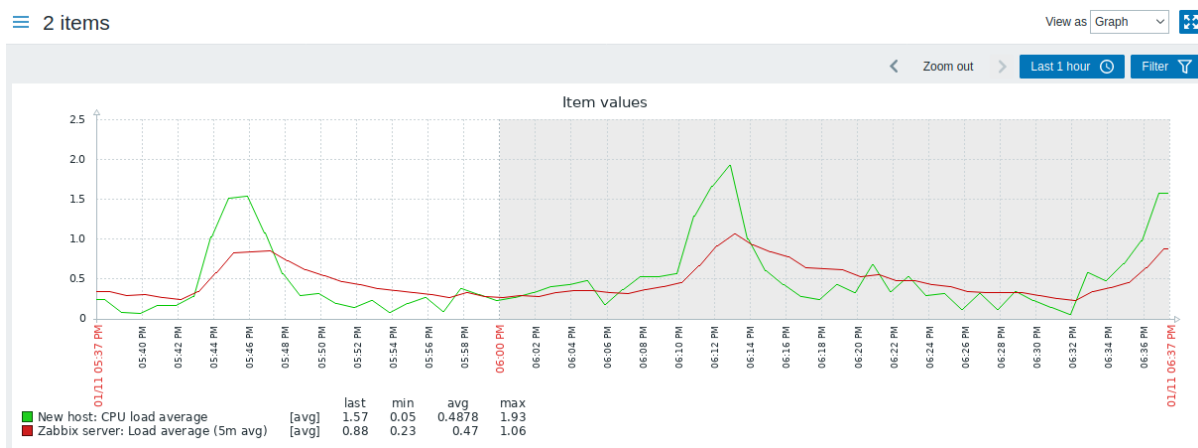
- Vá para Monitoramento → Últimos dados
- Use o filtro para exibir os itens que você deseja
- Marque as caixas de seleção dos itens que deseja representar graficamente
- Clique nos botões Exibir gráfico empilhado ou Exibir gráfico

Latest data

<input type="checkbox"/>	Host ▲	Name	Last check	Last value
<input checked="" type="checkbox"/>	New host	CPU load average	05/24/2021 10:46:5...	0.86
<input type="checkbox"/>	Zabbix server	Load average (1m avg)	05/24/2021 10:47:1...	0.73
<input type="checkbox"/>	Zabbix server	Load average (15m avg)	05/24/2021 10:47:1...	0.93
<input checked="" type="checkbox"/>	Zabbix server	Load average (5m avg)	05/24/2021 10:47:1...	0.93

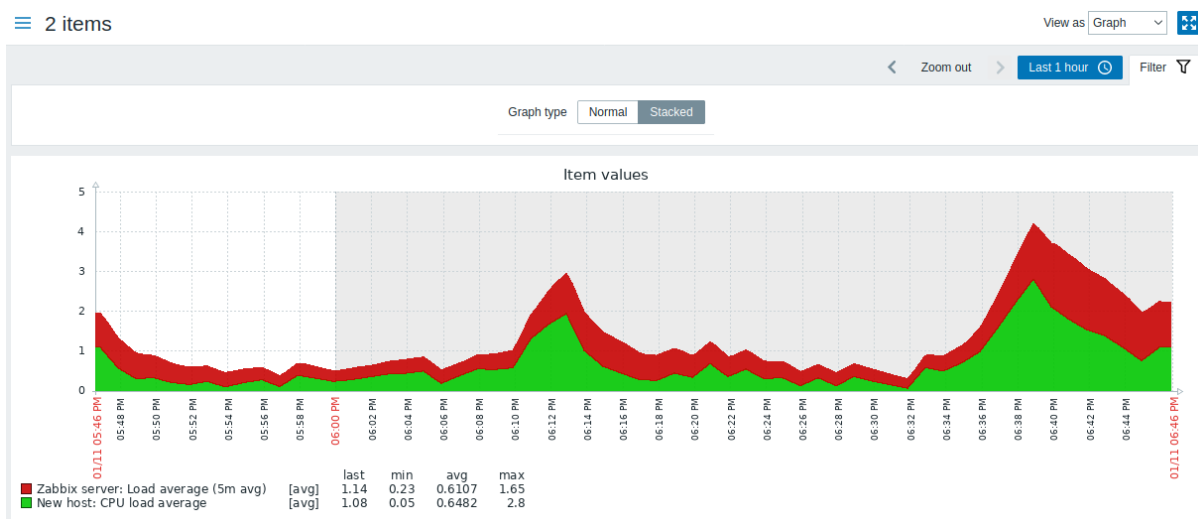
2 selected [Display stacked graph](#) [Display graph](#)

Seu gráfico é criado instantaneamente:



Observe que, para evitar a exibição de muitas linhas no gráfico, apenas o valor médio para cada item é exibido (linhas de valor mínimo/máximo não são exibido). Triggers e informações de trigger não são exibidas no gráfico.

Na janela do gráfico criado você tem o **período de tempo seletor** disponível e a possibilidade de mudar do gráfico de linhas "normal" para um empilhado (e de volta).



4 Agregação em gráficos

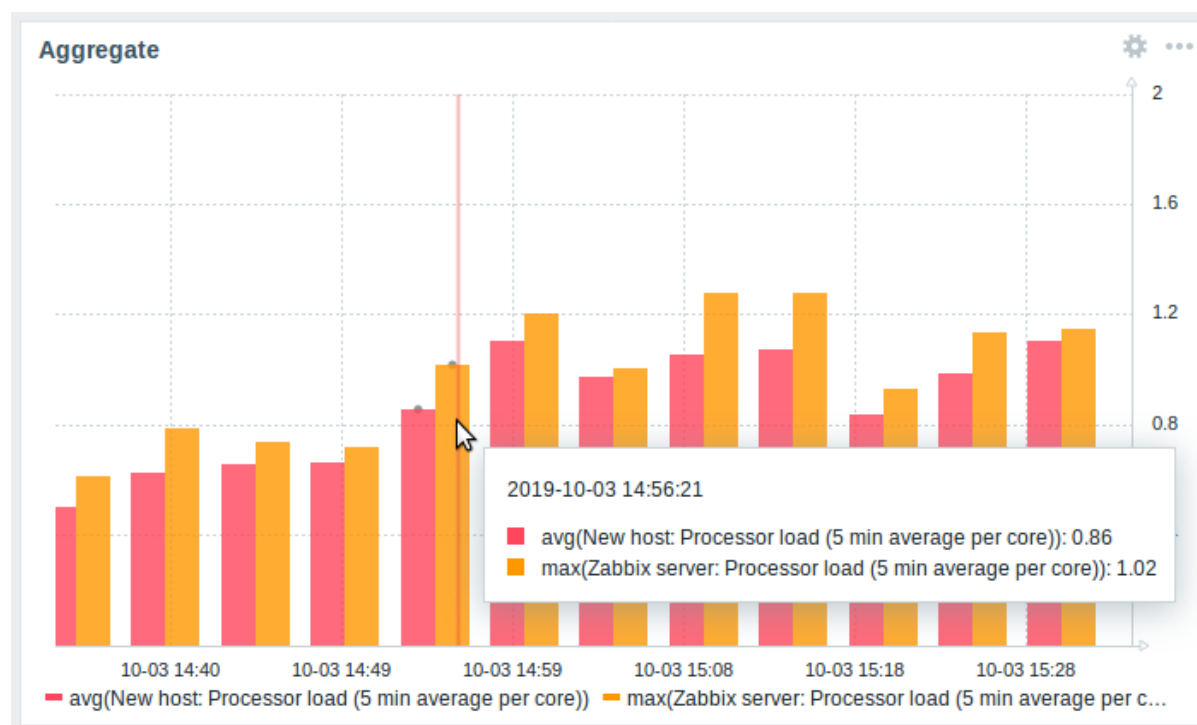
Visão geral

As funções de agregação, disponíveis no widget de gráfico do dashboard, permitem exibir um valor agregado para o intervalo escolhido (5 minutos, uma hora, um dia), em vez de todos os valores.

As opções de agregação são as seguintes:

- min
- máximo
- média
- contar
- soma
- primeiro (primeiro valor exibido)
- último (último valor exibido)

O uso mais interessante da agregação de dados é a possibilidade de criar boas comparações lado a lado de dados para algum período:



Ao passar o mouse sobre um ponto no tempo no gráfico, a data e a hora são exibidos, além dos itens e seus valores agregados. Os itens são exibidos entre parênteses, prefixado pela função de agregação usada. Observe que esta é a data e hora do ponto no gráfico, não de os valores reais.

Configuração

As opções de agregação estão disponíveis nas configurações do conjunto de dados quando configurando um **gráfico widget**.

Missing data ☐ None ☐ Connected ☐ T

Y-axis ☐ Left ☒ Right

Time shift

Aggregation function

Aggregation interval

Aggregate ☒ Each item ☐ Data set

Você pode escolher a função de agregação e o intervalo de tempo. Como os dados conjunto pode ser composto por vários itens,

existe também outra opção que permite mostrar dados agregados para cada item separadamente ou para todos os itens do conjunto de dados como um valor agregado.

Casos de uso

Contagem média de solicitações para o servidor Nginx

Veja a contagem média de solicitações por segundo por dia para o servidor Nginx:

- adicione o item de contagem de solicitações por segundo ao conjunto de dados
- selecione a função agregada `avg` e especifique o intervalo `1d`
- é exibido um gráfico de barras, onde cada barra representa a média número de solicitações por segundo por dia

Espaço mínimo semanal em disco entre clusters

Visualize o menor espaço em disco entre os clusters ao longo de uma semana.

- adicione ao conjunto de dados: `hosts cluster*`, chave `"Free disk space on /dados"`
- selecione a função agregada `min` e especifique o intervalo `1w`
- é exibido um gráfico de barras, onde cada barra representa o disco mínimo espaço por semana para cada volume `/data` do cluster

2 Mapas de rede

[comentário]: # (tags: mapa)

Overview

If you have a network to look after, you may want to have an overview of your infrastructure somewhere. For that purpose, you can create maps in Zabbix - of networks and of anything you like.

All users can create network maps. The maps can be public (available to all users) or private (available to selected users).

Proceed to [configuring a network map](#).

1 Configurando um mapa de rede

Visão geral

Configurar um mapa no Zabbix requer que você primeiro crie um mapa definindo seus parâmetros gerais e então você começa a preencher o real mapa com elementos e seus links.

Você pode preencher o mapa com elementos que são um host, um grupo de hosts, um gatilho, uma imagem ou outro mapa.

Os ícones são usados para representar os elementos do mapa. Você pode definir as informações que será exibido com os ícones e definir que os problemas recentes são exibido de forma especial. Você pode vincular os ícones e definir informações a serem exibidas nos links.

Você pode adicionar URLs personalizados para serem acessíveis clicando nos ícones. portanto você pode vincular um ícone de host a propriedades de host ou um ícone de mapa a outro mapa.

Os mapas são gerenciados no Monitoramento → [Mapas](#), onde eles podem ser configurados, gerenciados e visualizados. Na visualização de monitoramento, você pode clicar nos ícones e aproveitar os links para alguns scripts e URLs.

Os mapas de rede são baseados em gráficos vetoriais (SVG) desde o Zabbix 3.4.

Mapas públicos e privados

Todos os usuários do Zabbix (incluindo usuários não administradores) podem criar mapas de rede. Os mapas têm um proprietário - o usuário que os criou. Os mapas podem ser tornados públicos ou privado.

- Mapas Públicos são visíveis para todos os usuários, embora para vê-los o usuário deve ter acesso de leitura a pelo menos um elemento do mapa. Os mapas públicos podem ser editado caso um usuário/grupo de usuários tenha permissões de leitura/gravação para este mapa e pelo menos permissões de leitura para todos os elementos do mapa correspondente incluindo gatilhos nos links.
- Mapas Privados são visíveis apenas para seus proprietários e usuários/usuários agrupa o mapa é [compartilhado](#) com pelo proprietário. Usuários comuns (não superadministradores) só podem compartilhar com os grupos e usuários dos quais são membros. Os usuários de nível de administrador podem ver mapas privados independentemente de ser o proprietário ou pertencer ao lista de usuários compartilhada. Mapas privados podem ser editados pelo proprietário do mapa e caso um usuário/grupo de usuários tenha permissões de leitura/gravação para este mapear e pelo menos ler permissões para todos os elementos do mapa correspondente incluindo gatilhos nos links.

Os elementos do mapa para os quais o usuário não tem permissão de leitura são exibido com um ícone acinzentado e todas as informações textuais no elemento está oculto. No entanto, o rótulo do acionador é visível mesmo se o usuário não tem permissão para o gatilho.

Para adicionar um elemento ao mapa o usuário também deve ter pelo menos lido permissão para isso.

Criando um mapa

Para criar um mapa, faça o seguinte:

- Vá para Monitoramento → Mapas
- Vá para a vista com todos os mapas
- Clique em Criar mapa

Você também pode usar os botões Clone e Clone completo no formulário de configuração de um mapa existente para criar um novo mapa. Clicando em Clone manterá os atributos gerais de layout do mapa original, mas nenhum elemento. Clone completo manterá ambos os atributos gerais de layout e todos os elementos do mapa original.

A guia **Mapa** contém atributos gerais do mapa:

Map
Sharing

* Owner
Admin (Zabbix Administrator) X

* Name
Local network

* Width
680

* Height
600

Background image
No image v

Automatic icon mapping
<manual> v
[show icon mappings](#)

Icon highlight
☒

Mark elements on trigger status change
☒

Display problems
Expand single problem
Number of problems
Number of p

Advanced labels
☒

Host group label type
Label v

Host label type
Label v

Trigger label type
Status only v

Map label type
Label v

Image label type
Nothing v

Map element label location
Bottom v

Problem display
All v

Minimum severity
Not classified
Information
Warning
Average
High

Show suppressed problems
☐

URLs

Name
URL

Latest data
https://localhost/zabbix/latest.php

Add
.....

Add
Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Atributos gerais do mapa:

Parâmetro	Descrição
Proprietário	Nome do proprietário do mapa.
Nome	Nome do mapa exclusivo.
Largura	Largura do mapa em pixels.
Altura	Altura do mapa em pixels.
Imagem de fundo	Usar imagem de fundo: Sem imagem - sem imagem de fundo (fundo branco) Imagem - imagem selecionada para ser usada como imagem de fundo. Nenhum dimensionamento é executado. Você pode usar um mapa geográfico ou qualquer outra imagem para aprimorar seu mapa.
Mapeamento automático de ícones	Você pode configurar para usar um mapeamento automático de ícones, configurado em Administração → Geral → Mapeamento de ícones. O mapeamento de ícones permite mapear determinados ícones em relação a determinados campos de inventário do host.
Ícone em destaque	Se você marcar esta caixa, os elementos do mapa receberão destaque. Os elementos com um acionador ativo receberão um plano de fundo redondo, na mesma cor do acionador de gravidade mais alta. Além disso, uma linha verde grossa será exibida ao redor do círculo, se todos os problemas forem reconhecidos. Os elementos com status "desativado" ou "em manutenção" terão um fundo quadrado, cinza e laranja, respectivamente. Veja também: Visualizando mapas
Marcar elementos na alteração do status do acionador	Uma alteração recente do status do acionador (problema ou resolução recente) será destacada com marcadores (triângulos vermelhos apontando para dentro) nos três lados do ícone do elemento que não tem rótulo. Os marcadores são exibidos por 30 minutos.
Exibir problemas	Selecione como os problemas são exibidos com um elemento de mapa: Expandir problema único - se houver apenas um problema, o nome do problema será exibido. Caso contrário, o número total de problemas é exibido. Número de problemas - o número total de problemas é exibido Número de problemas e expandir o mais crítico - o nome do mais crítico problema e o número total de problemas é exibido. 'Mais crítico' é determinado com base na gravidade do problema e, se igual, ID do evento do problema (ID maior ou problema posterior exibido primeiro). Para um elemento de mapa de acionamento, ele é baseado na gravidade do problema e, se for igual, na posição do acionador na lista de acionadores. No caso de vários problemas do mesmo gatilho, será exibido o mais recente.
Rótulos avançados	Se você marcar esta caixa, poderá definir tipos de rótulos separados para tipos de elementos separados.
Tipo de rótulo do elemento do mapa	Tipo de rótulo usado para elementos do mapa: Rótulo - rótulo do elemento do mapa endereço IP - endereço IP Nome do elemento - elemento nome (por exemplo, nome do host) Somente status - somente status (OK ou PROBLEMA) Nada - nenhum rótulo é exibido
Local do rótulo do elemento do mapa	Local do rótulo em relação ao elemento do mapa: Bottom - abaixo do elemento do mapa Left - à esquerda Right - à direita Top - acima do elemento do mapa

Parâmetro	Descrição
Exibição do problema	Exibir a contagem de problemas como: All - a contagem total de problemas será exibida Separated - a contagem de problemas não confirmada será exibida separada como um número da contagem total de problemas Unacknowledged only - somente a contagem de problemas não confirmados será exibida
Gravidade mínima do gatilho	Problemas abaixo do nível de gravidade mínimo selecionado não serão exibidos no mapa. Por exemplo, com Aviso selecionado, as alterações com gatilhos de nível Informações e Não classificado não serão refletidas no mapa. Este parâmetro é suportado a partir do Zabbix 2.2.
Mostrar problemas suprimidos	Marque a caixa de seleção para exibir problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.
URLs	URLs para cada tipo de elemento podem ser definidos (com um rótulo). Eles serão exibidos como links quando um usuário clicar no elemento no modo de visualização do mapa. As macros podem ser usadas em nomes e valores de URL de mapa. Para obter uma lista completa, consulte macros com suporte e procure por 'mapear nomes e valores de URL'.

Compartilhamento

A guia **Compartilhamento** contém o tipo de mapa e as opções de compartilhamento (grupos de usuários, usuários) para mapas privados:

Parâmetro	Descrição
Tipo	Selecione o tipo de mapa: Privado - o mapa é visível apenas para grupos de usuários e usuários selecionados Público - o mapa é visível para todos
Lista de compartilhamentos de grupos de usuários	Selecione os grupos de usuários aos quais o mapa é acessível. Você pode permitir acesso somente leitura ou leitura-gravação.
Lista de compartilhamentos de usuários	Selecione os usuários aos quais o mapa é acessível. Você pode permitir acesso somente leitura ou leitura-gravação.

Ao clicar em Adicionar para salvar este mapa, você criou um mapa vazio com um nome, dimensões e certas preferências. Agora você precisa adicionar alguns elementos. Para isso, clique em Construtor na lista de mapas para abrir a área editável.

Adicionando elementos

Para adicionar um elemento, clique em Adicionar ao lado de Elemento do mapa. O novo elemento aparecerá no canto superior esquerdo do mapa. Arraste e solte em qualquer lugar você gosta.

Observe que com a opção Grid "On", os elementos sempre se alinharão ao grade (você pode escolher vários tamanhos de grade no menu suspenso, também ocultar/mostrar a grade). Se você quiser colocar elementos em qualquer lugar sem alinhamento, gire a opção "Desativar". (Elementos aleatórios podem mais tarde ser alinhados novamente ao grade com o botão Alinhar elementos do mapa.)

Agora que você tem alguns elementos no lugar, você pode querer começar diferenciando-os dando nomes, etc. Ao clicar no elemento, um formulário é exibido e você pode definir o tipo de elemento, dar um nome, escolher um ícone diferente, etc.

Map element: [Add](#) / [Remove](#) Shape: [Add](#) / [Remove](#) Link: [Add](#) / [Remove](#) Expand macros: [Off](#) Grid: [Shown](#) / [On](#) 50x50 [Align map elements](#) [Update](#)

Y X: 50 100 150 200 250 300 350 400 450 500 550 600 650 700

(MAP.NAME)

New element

(HOST.NAME)
(HOST.CONN)

Map element

Type Host

Label

Label location Default

* Host [Select](#)

Tags

And/Or Or

Contains [Remove](#)

[Add](#)

Automatic icon selection ☐

Icons

Default Server_(64)

Problem Default

Maintenance Default

Disabled Default

Coordinates X Y

URLs

Name	URL	Action
		Remove

Atributos do elemento do mapa:

Parâmetro	Descrição
Tipo	<p>Tipo do elemento:</p> <p>Host - ícone que representa o status de todos os acionadores do host selecionado</p> <p>Map - ícone que representa o status de todos os elementos de um mapa</p> <p>Trigger - ícone que representa o status de um ou mais acionadores</p> <p>Host group - ícone que representa o status de todos os acionadores de todos os hosts pertencentes ao grupo selecionado</p> <p>Imagem - um ícone, não vinculado a nenhum recurso</p>
Rótulo	<p>Rótulo do ícone, qualquer string.</p> <p>Macros e strings de várias linhas podem ser usados.</p> <p>Expressão macros são suportadas neste campo, mas apenas com Funções avg, last, min e max, com tempo como parâmetro (por exemplo, {?avg(/host/key, 1h)}).</p> <p>Para uma lista completa de funções suportadas macros, consulte macros suportados e procure por 'rótulos de elementos do mapa'.</p>
Local do rótulo	<p>Local do rótulo em relação ao ícone:</p> <p>Padrão - local do rótulo padrão do mapa</p> <p>Bottom - abaixo do ícone</p> <p>Left - para o esquerda</p> <p>Direita - à direita</p> <p>Topo - acima do ícone</p>
Host	<p>Digite o host se o tipo de elemento for 'Host'. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um host oferecerá uma lista suspensa de hosts correspondentes.</p> <p>Role para baixo para selecionar. Clique em 'x' para remover o selecionado.</p>
Map	<p>Selecione o mapa, se o tipo de elemento for 'Map'.</p>
Triggers	<p>Se o tipo de elemento for 'Trigger', selecione um ou mais triggers no campo New triggers abaixo e clique em Add.</p> <p>A ordem dos triggers selecionados pode ser alterada, mas apenas dentro do mesma gravidade dos gatilhos. A seleção de vários acionadores também afeta a resolução da macro {HOST.*} nos modos de construção e visualização.</p> <p>// 1 No modo de construção// as primeiras macros {HOST.*} exibidas serão resolvidas dependendo do primeiro acionador na lista (com base na gravidade do acionador).</p> <p>// 2 O modo de visualização// depende do parâmetro Display problems nos atributos gerais do mapa.</p> <p>* Se o modo Expandir problema único for escolhido, as primeiras macros {HOST.*} exibidas serão resolvidas dependendo do último acionador de problema detectado (não importando a gravidade) ou do primeiro acionador na lista (caso nenhum problema seja detectado);</p> <p>* Se Número de problemas e expandir o modo mais crítico for escolhido, as primeiras macros {HOST.*} exibidas serão resolvidas dependendo da gravidade do acionador.</p>
Host group	<p>Digite o host group se o tipo de elemento for 'Host group'. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes. Role para baixo para selecionar. Clique em 'x' para remover o selecionado.</p>

Parâmetro	Descrição
Tags	<p>Especifique as tags para limitar o número de problemas exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas. Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclua os nomes de tags especificados</p> <p>Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
Ou - suficiente se uma condição for atendida</p> <p>Este campo está disponível para tipos de elementos de host e grupo de host.</p>
Seleção automática de ícones	Neste caso, um mapeamento de ícones será usado para determinar qual ícone exibir.
Ícones	Você pode optar por exibir ícones diferentes para o elemento nestes casos: padrão, problema, manutenção, desabilitado.
Coordenada X	Coordenada X do elemento do mapa.
Coordenada Y	Coordenada Y do elemento do mapa.
URLs	<p>URLs específicas do elemento podem ser definidas para o elemento. Eles serão exibidos como links quando um usuário clicar no elemento no modo de visualização do mapa. Se o elemento tiver seus próprios URLs e houver URLs de nível de mapa para seu tipo definido, eles serão combinados no mesmo menu.</p> <p>As macros podem ser usadas em nomes e valores de elementos de mapa. Para obter uma lista completa, consulte macros com suporte e procure por 'mapear nomes e valores de URL'.</p>

::: não importante Os elementos adicionados não são salvos automaticamente. Se você navegar para fora da página, todas as alterações podem ser perdidas.

Portanto, é uma boa ideia clicar no botão **Atualizar** na parte superior Esquina direita. Uma vez clicado, as alterações são salvas, independentemente do que você escolha no pop-up a seguir.

As opções de grade selecionadas também são salvas com cada mapa. :::

Selecionando elementos

Para selecionar elementos, selecione um e, em seguida, mantenha pressionado Ctrl para selecionar o outras.

Você também pode selecionar vários elementos arrastando um retângulo na área editável e selecionando todos os elementos nela.

Depois de selecionar mais de um elemento, a forma de propriedade do elemento muda para o modo de atualização em massa para que você possa alterar os atributos de elementos de uma só vez. Para fazer isso, marque o atributo usando a caixa de seleção e insira um novo valor para ele. Você pode usar macros aqui (por exemplo, {HOST.NAME} para o rótulo do elemento).

Map element: [Add](#) / [Remove](#) Shape: [Add](#) / [Remove](#) Link: [Add](#) / [Remove](#) Expand macros: [Off](#) Grid: [Shown](#) / [On](#) 50x50 [Align map elements](#) [Update](#)

Mass update elements

Selected elements	Type	Name
<input checked="" type="checkbox"/>	Host	My host
<input checked="" type="checkbox"/>	Host	vcenter.zabbix.lan

☒ Label

☒ Label location

☐ Automatic icon selection ☐

☐ Icon (default)

☐ Icon (problem)

☐ Icon (maintenance)

☐ Icon (disabled)

Elementos de ligação

Depois de colocar alguns elementos no mapa, é hora de começar a vincular eles. Para vincular dois elementos, você deve primeiro selecioná-los. Com os elementos selecionado, clique em Adicionar ao lado de Link.

Com um link criado, o formulário de elemento único agora contém um seção Links. Clique em Editar para editar os atributos do link.

Map element: [Add / Remove Shape](#): [Add / Remove Link](#): [Add / Remove Expand macros](#): Off Grid: [Shown / On](#) 50x50 [Align map elements](#) [Update](#)

X: Y:

50	100	150	200	250	300	350	400	450	500	550	600	650	700
----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

{MAP.NAME}

New element

100Mbps

{HOST.NAME}
{HOST.CONN}

Map element

Type: Host

Label: New element

Label location: Default

* Host: My host X Select

Application: Select

Automatic icon selection: ☐

Icons

Default	Server_(96)
Problem	Default
Maintenance	Default
Disabled	Default

Coordinates X: 89 Y: 127

URLs

Name	URL	Action
		Remove

Add

Apply Remove Close

Links

Element name	Link indicators	Action
vcenter.zabbix.lan		Edit

Label: 100Mbps

Connect to: vcenter.zabbix.lan

Type (OK): Bold line

Color (OK): 00CC00

Link indicators

Trigger	Type	Color	Action
Add			

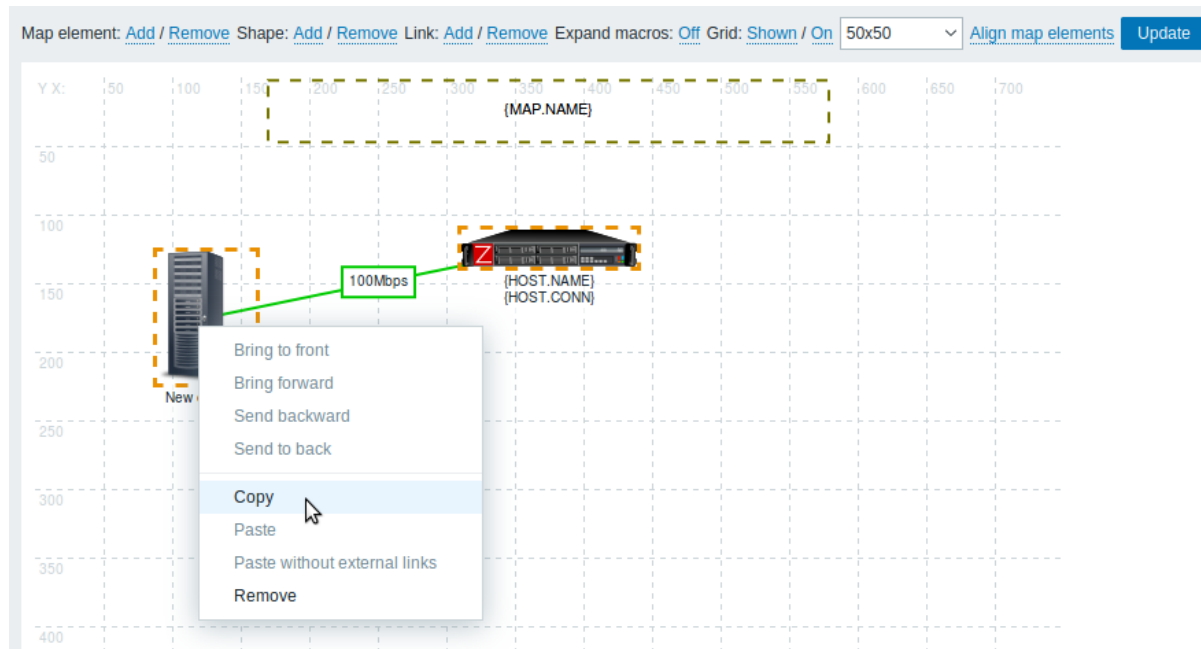
Apply Remove Close

Parâmetro	Descrição
Rótulo	Rótulo que será renderizado no topo do link. Expressão macros são suportadas neste campo, mas apenas com avg, last, Funções min e max, com tempo como parâmetro (por exemplo, {?avg(/host/key,1h)}).
Connect to Digite (OK)	O elemento ao qual o link se conecta. Estilo de link padrão: Linha - linha única Linha em negrito - linha em negrito Ponto - pontos ** Linha tracejada** - linha tracejada
Cor (OK)	Cor padrão do link.
Indicadores de link	Lista de gatilhos vinculados ao link. Caso uma trigger tenha o status PROBLEM, seu estilo é aplicado ao link.

Movendo e copiando e colando elementos

Vários elementos selecionados podem ser **movidos** para outro lugar no mapa clicando em um dos elementos selecionados, mantendo pressionado o mouse botão e movendo o cursor para o local desejado.

Um ou mais elementos podem ser **copiados** selecionando os elementos e, em seguida, clicando em um elemento selecionado com o botão direito do mouse e selecionando Copiar do menu.



Para colar os elementos, clique em uma área do mapa com o botão direito do mouse e selecione Colar no menu. O Colar sem links externos opção irá colar os elementos retendo apenas os links que estão entre os elementos selecionados.

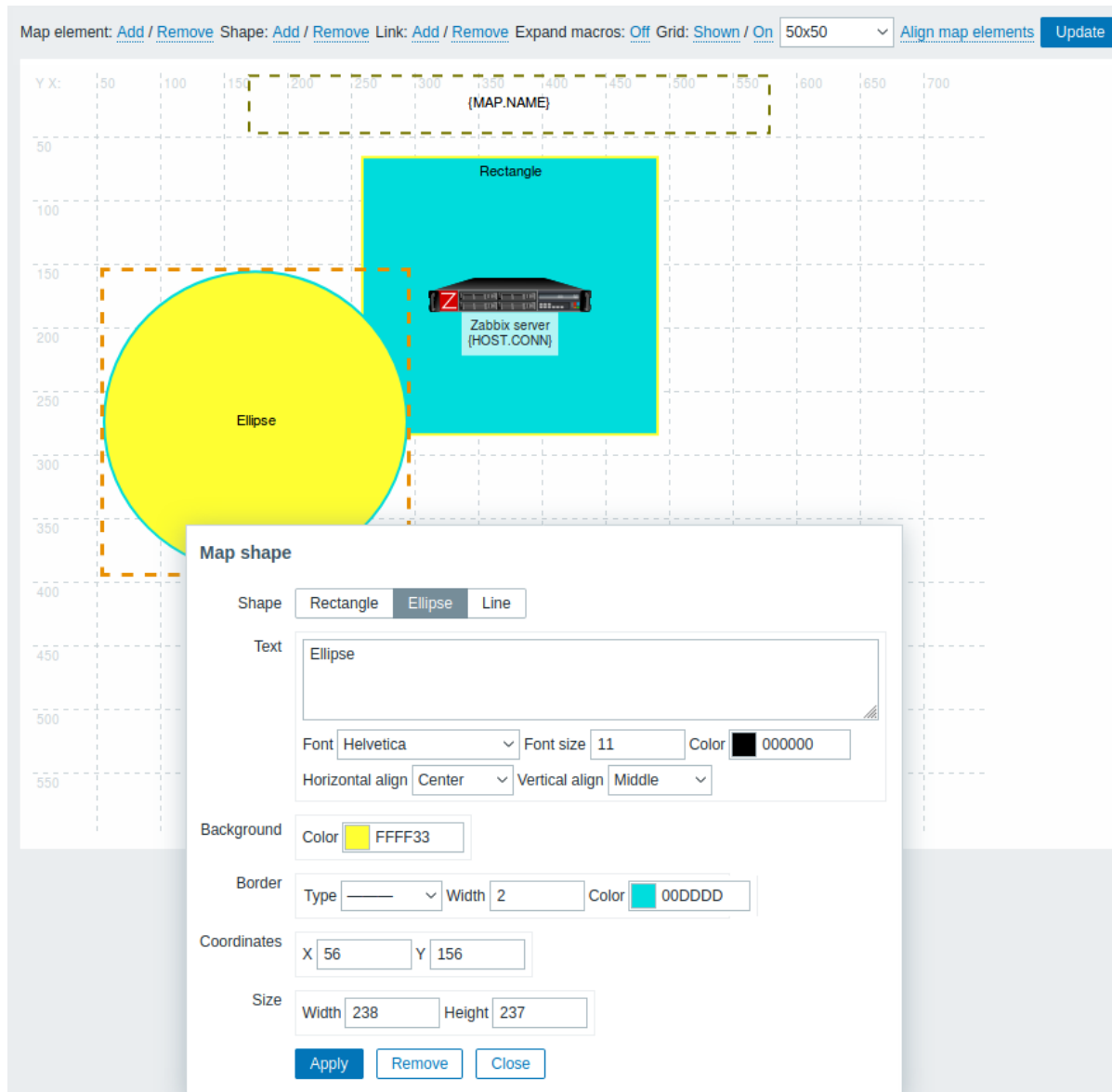
Copiar e colar funciona na mesma janela do navegador. Atalhos do teclado não são suportados.

Adicionando formas

Além dos elementos do mapa, também é possível adicionar algumas formas. Formas não são elementos de mapa; são apenas uma representação visual. Por exemplo, uma forma de retângulo pode ser usada como plano de fundo para agrupar alguns anfitriões. Formas de retângulo e elipse podem ser adicionadas.

Para adicionar uma forma, clique em Adicionar ao lado de Forma. A nova forma aparecerá no canto superior esquerdo do mapa. Arraste e solte onde quiser.

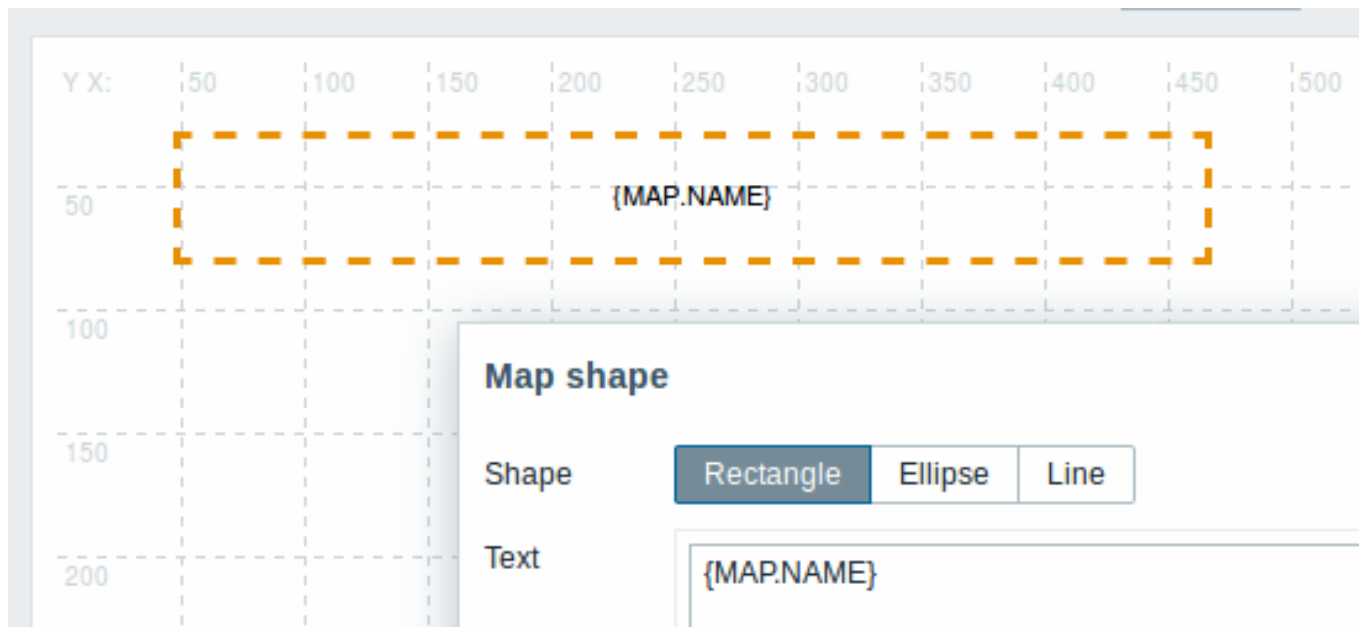
Uma nova forma é adicionada com cores padrão. Ao clicar na forma, um formulário é exibido e você pode personalizar a aparência de uma forma, adicionar texto, etc.



Para selecionar formas, selecione uma e, em seguida, mantenha pressionado Ctrl para selecionar o outras. Com várias formas selecionadas, as propriedades comuns podem ser de massa atualizado, da mesma forma que com elementos.

O texto pode ser adicionado nas formas. Expressão **macros** são compatíveis com o texto, mas apenas com as funções avg, last, min e max, com time como parâmetro (por exemplo, `{?avg(/host/key,1h)}`).

Para exibir texto, apenas a forma pode ficar invisível removendo o borda da forma (selecione 'Nenhum' no campo Border). Por exemplo, tome observação de como a macro `{MAP.NAME}`, visível na captura de tela acima, é na verdade uma forma de retângulo com texto, que pode ser visto ao clicar em a macro:



{MAP.NAME} resolve para o nome do mapa configurado ao visualizar o mapa.

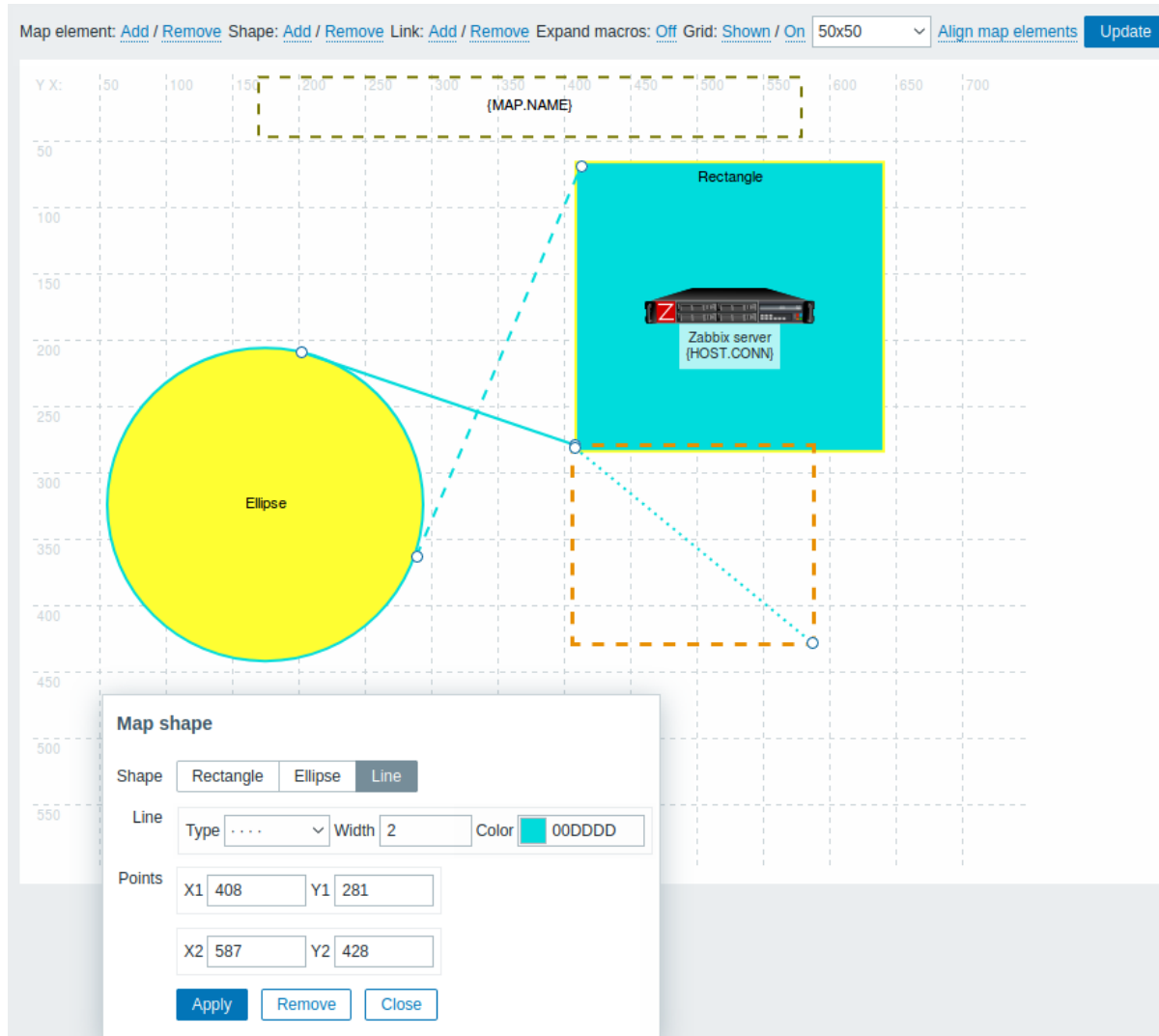
Se hiperlinks forem usados no texto, eles se tornarão clicáveis ao visualizar o mapa.

A quebra de linha para texto está sempre “ativada” nas formas. No entanto, dentro de um elipse, as linhas são quebradas como se a elipse fosse um retângulo. A quebra de palavras não está implementada, então palavras longas (palavras que não se encaixam a forma) não são encapsulados, mas são mascarados (página do construtor) ou recortados (outras páginas com mapas).

Adicionando linhas

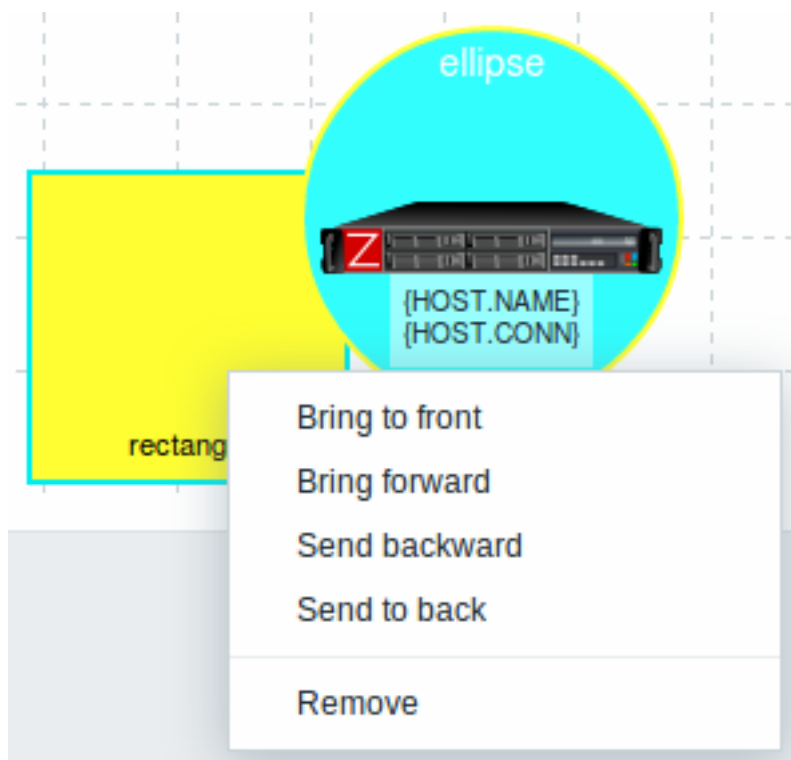
Além das formas, também é possível adicionar algumas linhas. As linhas podem ser usado para vincular elementos ou formas em um mapa.

Para adicionar uma linha, clique em Adicionar ao lado de Forma. Uma nova forma aparecerá em no canto superior esquerdo do mapa. Selecione-o e clique em Linha no formulário de edição para alterar a forma em uma linha. Em seguida, ajuste a linha propriedades, como tipo de linha, largura, cor, etc.



Ordenando formas e linhas

Para trazer uma forma na frente da outra (ou vice-versa) clique no forma com o botão direito do mouse trazendo o menu de forma do mapa.



2 Elementos do grupo de hosts

Visão geral

Esta seção explica como adicionar um elemento do tipo “Host group” quando configurando um mapa de rede.

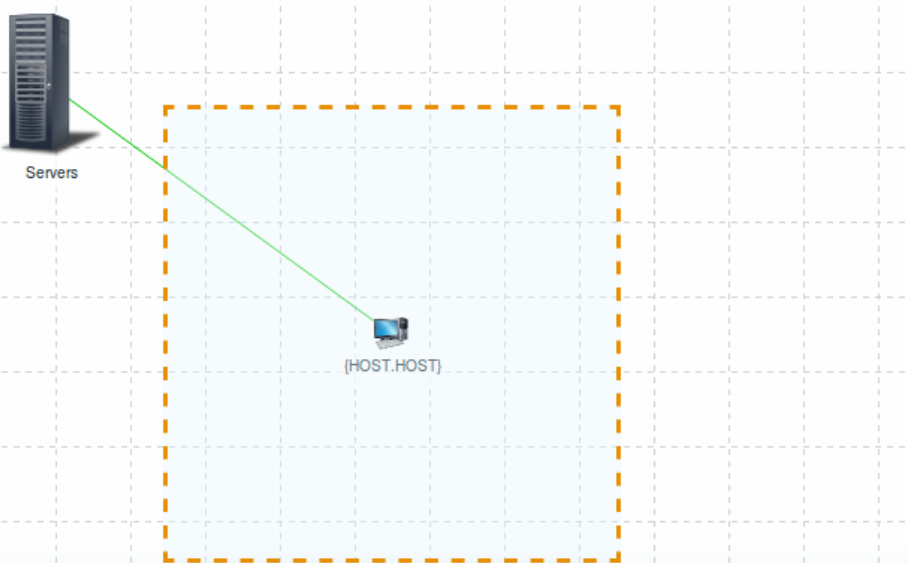
Configuração

Map element: [Add](#) / [Remove](#) Shape: [Add](#) / [Remove](#) Link: [Add](#) / [Remove](#) Expand macros: [Off](#) Grid: [Shown](#) / [On](#) 50x50 [Align map elements](#)

Y X: 50 100 150 200 250 300 350 400

Local network 2

50 100 150 200 250 300 350 400



Map element

Type

Host group

Show

Host group

Host group elements

Area type

Fit to map

Custom size

Area size Width

300

 Height

300

Placing algorithm

Grid

Label

{HOST.HOST}

Label location

Default

* Host group

Linux servers

Select

Application

Select

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Esta tabela consiste em parâmetros típicos para o tipo de elemento Host group:

Parâmetro	Descrição
Tipo	Selecione o tipo do elemento: Host group - ícone que representa o status de todos os triggers de todos os hosts pertencentes ao grupo selecionado

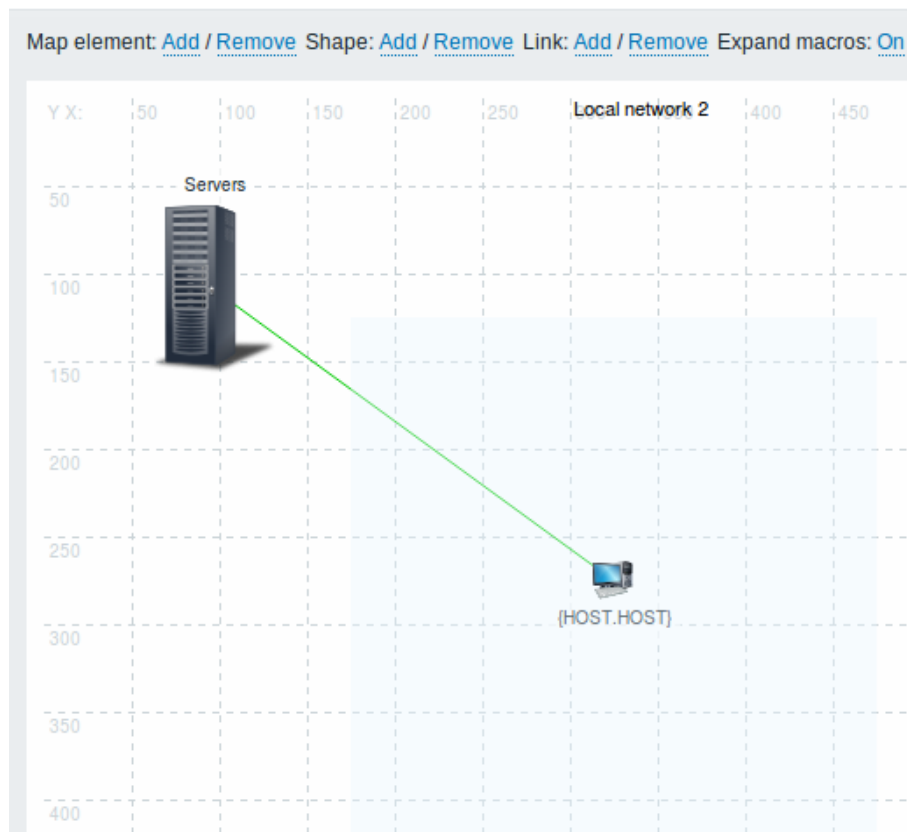
531

Parâmetro	Descrição
Mostrar	Mostrar opções: Grupo de hosts - selecionar esta opção resultará em um único ícone exibindo informações correspondentes sobre determinado grupo de hosts Elementos do grupo de hosts - selecionar esta opção resultará como vários ícones exibindo informações correspondentes sobre cada elemento (host) de um determinado grupo de hosts
Tipo de área	Esta configuração está disponível se o parâmetro "Host group elements" for selecionado: Ajustar ao mapa - todos os elementos do grupo de hosts são colocados igualmente dentro do mapa **Tamanho personalizado** - uma configuração manual da área do mapa para todos os elementos do grupo de hosts a serem exibidos
Tamanho da área	Esta configuração está disponível se o parâmetro "Elementos do grupo de host" e o parâmetro "Tipo de área" forem selecionados: Largura - valor numérico a ser inserido para especificar a largura da área do mapa **Altura** - valor numérico a ser inserido para especificar a altura da área do mapa
Algoritmo de colocação	Grid - única opção disponível de exibir todos os elementos do grupo de hosts
Rótulo	Rótulo do ícone, qualquer string. Macros e strings de várias linhas podem ser usados em rótulos. Se o tipo do elemento do mapa for "Grupo de hosts" especificar certas macros tem um impacto na visualização do mapa exibindo informações correspondentes sobre cada host. Por exemplo, se a macro {HOST.IP} for usada, a visualização do mapa de edição exibirá apenas a macro {HOST.IP} enquanto a visualização do mapa incluirá e exibirá o endereço IP exclusivo de cada host

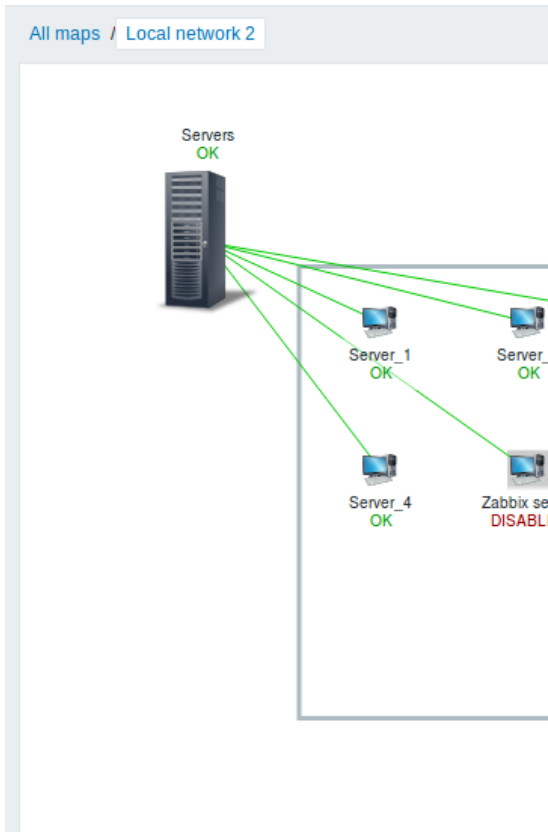
Visualizando elementos do grupo de hosts

Esta opção está disponível se a opção de exibição "Elementos do grupo de hosts" estiver escolhido. Ao selecionar "Elementos do grupo de hosts" como a opção show, você primeiro verá apenas um ícone para o grupo de hosts. No entanto, quando você salve o mapa e vá para a visualização do mapa, você verá que o mapa inclui todos os elementos (hosts) de um determinado grupo de hosts:

Network maps



Maps



Observe como a macro `{HOST.NAME}` é usada. Na edição do mapa, o nome da macro não é resolvido, enquanto na visualização do mapa todos os nomes exclusivos dos hosts são exibido.

3 Indicadores de links

Visão geral

Você pode atribuir alguns gatilhos a um **link** entre elementos em um mapa de rede. Quando esses gatilhos entram em um estado de problema, o link pode refletir isso.

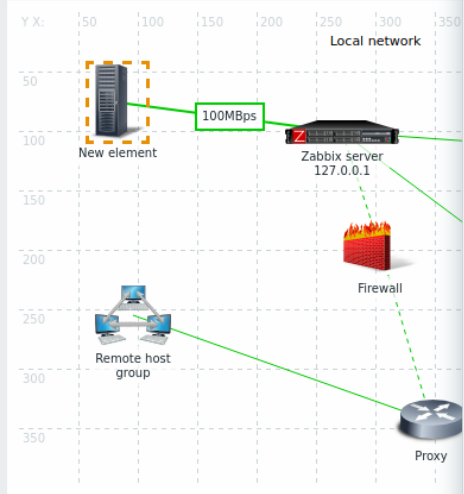
Ao configurar um link, você define o tipo e a cor do link padrão. Quando você atribui gatilhos a um link, você pode atribuir diferentes tipos de link e cores com esses gatilhos.

Se algum desses gatilhos entrar em um estado de problema, seu estilo de link e a cor será exibida no link. Então, talvez seu link padrão fosse uma linha verde. Agora, com o gatilho no estado do problema, seu link pode tornar-se vermelho em negrito (se você definiu assim).

Configuração

Para atribuir acionadores como indicadores de link, faça o seguinte:

- selecione um elemento do mapa
- clique em Editar na seção Links para obter o link apropriado
- clique em Adicionar no bloco Indicadores de link e selecione um ou mais gatilhos

Map element: [Add](#) / [Remove](#) Shape: [Add](#) / [Remove](#) Link: [Add](#) / [Remove](#)

Map element

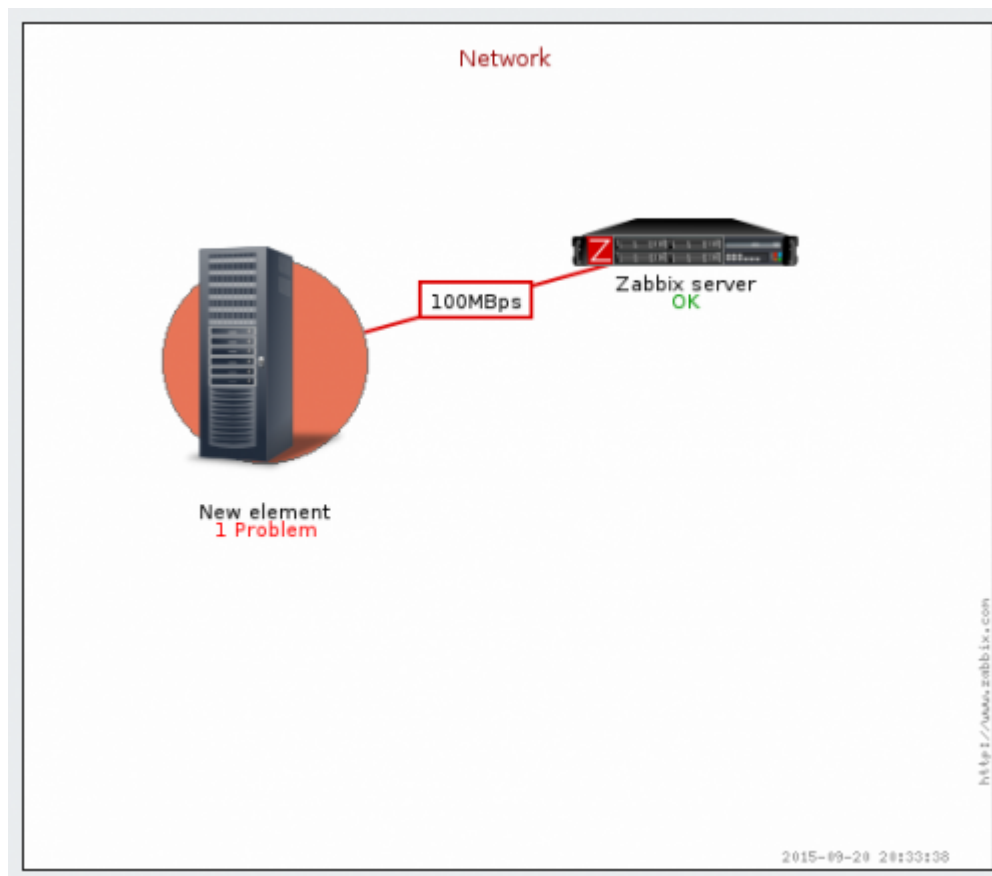
Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Os gatilhos adicionados podem ser vistos na lista Indicadores de link.

Você pode definir o tipo e a cor do link para cada acionador diretamente do Lista. Quando terminar, clique em Aplicar, feche o formulário e clique em Atualizar para salvar as alterações do mapa.

Tela

Em Monitoramento → Mapas a respectiva cor será exibida no link se o gatilho entrar em um estado de problema.



Note:

Se vários gatilhos entrarem em um estado de problema, o problema com a gravidade mais alta determinará o estilo e a cor do link. Se vários gatilhos com a mesma gravidade são atribuídos ao mesmo mapa link, aquele com o menor ID tem precedência. Observe também que:

1. Configurações de Severidade mínima do gatilho e Mostrar problema suprimido da configuração do mapa afetam quais problemas são levados em consideração.
2. No caso de gatilhos com vários problemas (múltiplos problemas geração), cada problema pode ter uma gravidade diferente do gatilho gravidade (alterada manualmente), pode ter tags diferentes (devido a macros), e pode ser reprimido.

3 Painéis

Painéis e seus widgets fornecem uma plataforma de visualização forte com tais ferramentas como gráficos modernos, mapas, apresentações de slides e muito mais.



4 Painéis de host

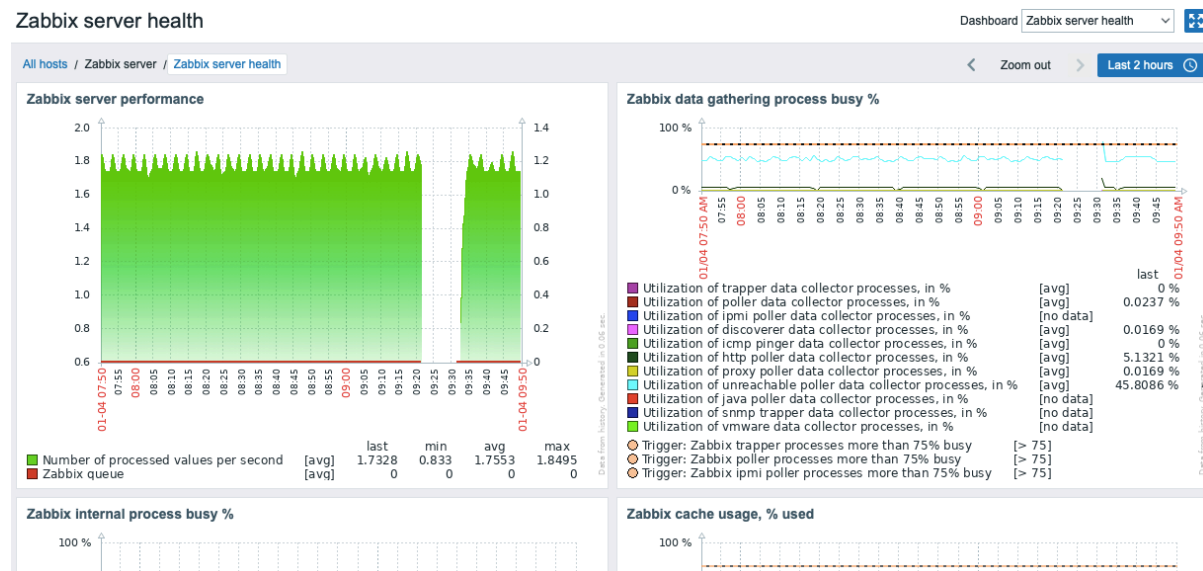
Visão geral

Os painéis do host são semelhantes a **global painéis**, no entanto, os painéis de host exibem dados apenas sobre o host. Hospedeiro painéis não têm proprietário.

Os painéis de host são configurados no **template** nível e, em seguida, são gerados para um host, uma vez que o template é vinculado ao hospedeiro. Widgets de painéis de host só podem ser copiados para painéis de host do mesmo molde. Widgets de painéis globais não podem ser copiados nos painéis do host.

Os painéis de host não podem ser configurados ou acessados diretamente no Monitoramento → **Dashboard** seção, que é reservada para painéis globais. As formas de acesso painéis de host estão listados abaixo nesta seção.

Zabbix server health



Ao visualizar os painéis do host, você pode alternar entre os painéis usando o menu suspenso no canto superior direito. Para mudar para Monitoring→Hosts, clique no link de navegação All hosts abaixo do nome do painel no canto superior esquerdo.

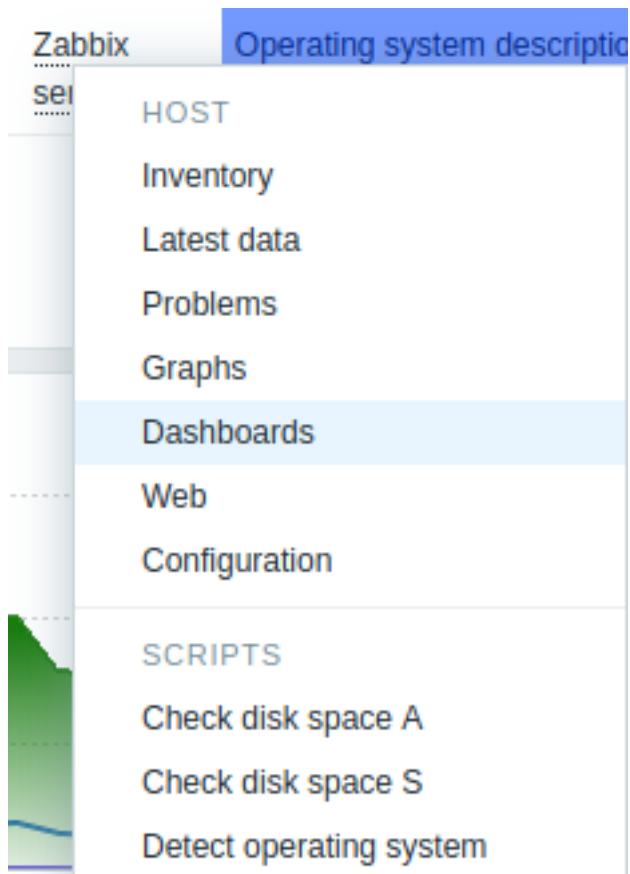
Os widgets dos painéis do host não podem ser editados.

Observe que os painéis de host costumavam ser telas de host antes do Zabbix 5.2. Ao importar um modelo mais antigo contendo telas, a importação de tela será ignorado.

Acessando painéis de host

O acesso aos painéis de host é fornecido:

- Do **anfitrião menu** que está disponível em muitos locais de front-end:
 - clique no nome do host e selecione Dashboards na menu suspenso



- Ao pesquisar um nome de host em **global search**:
 - clique no link Painéis fornecido nos resultados da pesquisa
- Ao clicar em um nome de host em Inventário → **Hosts**:
 - clique no link Painéis fornecido

8 Modelos

Visão geral

Um modelo é um conjunto de entidades que podem ser convenientemente aplicadas a vários hospedeiros.

As entidades podem ser:

- Itens
- gatilhos
- gráficos
- painéis
- regras de descoberta de baixo nível
- cenários da web

Como muitos hosts na vida real são idênticos ou bastante semelhantes, naturalmente segue que o conjunto de entidades (itens, gatilhos, gráficos,...) você criou para um host, pode ser útil para muitos. Claro, você poderia copiá-los para cada novo host, mas isso seria muito manual trabalhar. Em vez disso, com modelos, você pode copiá-los para um modelo e depois aplique o modelo a quantos hosts forem necessários.

Quando um modelo é vinculado a um host, todas as entidades (itens, gatilhos, gráficos,...) do modelo são adicionados ao host. Os modelos são atribuído diretamente a cada host individual (e não a um grupo de hosts).

Os modelos são frequentemente usados para agrupar entidades para serviços específicos ou aplicativos (como Apache, MySQL, PostgreSQL, Postfix...) e depois aplicado a hosts que executam esses serviços.

Outro benefício de usar modelos é quando algo precisa ser alterado para todos os anfitriões. Alterar algo no nível do modelo uma vez irá propagar a mudança para todos os hosts vinculados.

Assim, a utilização de templates é uma excelente forma de reduzir a carga de trabalho e simplificando a configuração do Zabbix.

Prossiga para **criando e configurando um template**.

9 Modelos prontos para uso

Visão geral

O Zabbix se esforça para fornecer uma lista crescente de ferramentas úteis prontas para uso **modelos**. Os modelos prontos para uso vêm pré-configurados e, portanto, são uma maneira útil de acelerar a implantação de monitoramento de trabalhos.

Os moldes estão disponíveis:

- Em novas instalações - em Configuração → Modelos;
- Se você estiver atualizando de versões anteriores, poderá encontrar esses templates no diretório `templates` do último download Versão Zabbix. Enquanto estiver em Configuração → Modelos você pode importá-los manualmente deste diretório.
- Também é possível baixar o template do [Zabbix git repositório](#) diretamente (certifique-se de que o template é compatível com seu Zabbix versão).

Por favor, use a barra lateral para acessar informações sobre um modelo específico tipos e requisitos de operação.

Veja também:

- [Importação de modelo](#)
- [Ligar um template](#)

Modelos padronizados para dispositivos de rede

Visão geral

Para fornecer monitoramento para dispositivos de rede, como switches e roteadores, criamos dois chamados modelos: para o dispositivo de rede em si (seu chassi basicamente) e para interface de rede.

Como os templates do Zabbix 3.4 para muitas famílias de dispositivos de rede são forneceu. Todos os modelos cobrem (onde possível obter esses itens de o dispositivo):

- Monitoramento de falhas do chassi (fontes de alimentação, ventiladores e temperatura, Estado geral)
- Monitoramento de desempenho do chassi (CPU e itens de memória)
- Coleta de inventário do chassi (números de série, nome do modelo, firmware versão)
- Monitoramento de interface de rede com IF-MIB e EtherLike-MIB (status da interface, carga de tráfego da interface, status duplex para Ethernet)

Estes modelos estão disponíveis:

- Em novas instalações - em Configuração → Modelos;
- Se você estiver atualizando de versões anteriores, poderá encontrar esses templates no diretório `templates` do último download Versão Zabbix. Enquanto estiver em Configuração → Modelos você pode importá-los manualmente deste diretório.

Se estiver importando os novos modelos prontos para uso, convém também atualize o regular global `@Network interfaces for discovery` expressão para:

```
O resultado é FALSE: ^Software Loopback Interface
O resultado é FALSE: ^([In]?[1L]oop[bB]ack[0-9._])*$
O resultado é FALSE: ^NULL[0-9.]*$
O resultado é FALSE: ^[1L]o[0-9.]*$
O resultado é FALSE: ^[sS]istema$
O resultado é FALSE: ^Nu[0-9.]*$
```

para filtrar loopbacks e interfaces nulas na maioria dos sistemas.

Dispositivos

Lista de famílias de dispositivos para os quais os modelos estão disponíveis:

```
[Nome do modelo|Fornecedor|Família de dispositivos|Modelos conhecidos|SO|MIBs usados|Tags| |-----|-----|-----|-----|
---| -|-----|-----|-----|-----| |Alcatel Timetra TiMOS SNMP|Alcatel|Alcatel Timetra|ALCATEL
SR 7750|TiMOS|TIMETRA-SYSTEM-MIB,TIMETRA-CHASSIS-MIB|Certified| |Brocade FC SNMP|Brocade|Comutadores Brocade
FC|Comutador Brocade 300 SAN-|SW-MIB,ENTITY-MIB|Desempenho, Falha| |Brocade_Foundry Stackable SNMP|Brocade|Brocade
ICX|Brocade ICX6610, Brocade ICX7250-48, Brocade ICX7450-48F|<|FOUNDRY-SN-AGENT-MIB, FOUNDRY-SN-STACKING-MIB|Certified|
|Brocade_Foundry SNMP não empilhável|Brocade, Foundry|Brocade MLX, Foundry|Brocade MLXe, Foundry FLS648, Foundry
FWSX424|<|FOUNDRY-SN-AGENT-MIB|Performance, Fault| |Cisco Catalyst 3750<modelo de dispositivo> SNMP|Cisco|Cisco
Catalyst 3750|Cisco Catalyst 3750V2-24FS, Cisco Catalyst 3750V2-24PS, Cisco Catalyst 3750V2-24TS, Cisco Catalyst SNMP,
```

Cisco Catalyst SNMP|<|CISCO-MEMORY-POOL-MIB, IF-MIB, EtherLike-MIB, SNMPv2-MIB, CISCO-PROCESS-MIB, CISCO-ENVMON-MIB, ENTITY-MIB|Certified| |Cisco IOS SNMP|Cisco|Cisco IOS ver > 12.2 3.5|Cisco C2950|IOS|CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB,CISCO-ENVMON-MIB|Certified| |Cisco IOS versões 12.0_3_T-12.2_3.5 SNMP|Cisco|Cisco IOS > 12.0 3 T e < 12.2 3.5|IOS|CISCO-PROCESS-MIB,CISCO-MEMORY-POOL-MIB ,CISCO-ENVMON-MIB|Certificado| |Cisco IOS antes de 12.0_3_T SNMP|Cisco|Cisco IOS < 12.0 3 T|IOS|OLD-CISCO-CPU-MIB,CISCO-MEMORY-POOL-MIB|Certified| |D-Link DES_DGS Switch SNMP|D-Link|DES/DGX switches|D-Link DES-xxxx/DGS-xxxx,DLINK DGS-3420-26SC|DLINK-AGENT-MIB,EQUIPMENT-MIB,ENTITY-MIB|Certificado| |D-Link DES 7200 SNMP|D-Link|DES-7xxx|D-Link DES 7206|ENTITY-MIB,MY-SYSTEM-MIB,MY-PROCESS-MIB,MY-MEMORY-MIB|Desempenho Interfaces de falha| |Dell Force S-Series SNMP|Dell|Dell Force S-Series|S4810|F10-S-SERIES-CHASSIS-MIB|Certified| |Extreme Exos SNMP|Extreme|Extreme EXOS|X670V-48x|EXOS|EXTREME-SYSTEM-MIB,EXTREME-SOFTWARE-MONITOR-MIB|Certified| |Huawei VRP SNMP|Huawei|Huawei VRP|S2352P-EI|ENTITY-MIB,HUAWEI-ENTITY-EXTENT-MIB|Certified| |Intel_Qlogic Infiniband SNMP|Intel/QLogic|Intel/QLogic Infiniband devices|Infiniband 12300|ICS-CHASSIS-MIB|Fault Inventory| |Juniper SNMP|Juniper|Modelos MX,SRX,EX|Juniper MX240, Juniper EX4200-24F|JUNOS|JUNIPER-MIB|Certified| |Mellanox SNMP|Mellanox|Dispositivos Mellanox Infiniband|SX1036|MLNX-OS|HOST-RESOURCES-MIB,ENTITY-MIB,ENTITY-SENSOR-MIB,MELLANOX-MIB|Certified| |MikroTik CCR<modelo de dispositivo> SNMP|MikroTik|MikroTik Cloud Core Routers (série CCR)|Modelos dedicados separados estão disponíveis para MikroTik CCR1009-7G-1C-1S+, MikroTik CCR1009-7G-1C-1S+PC, MikroTik CCR1009 -7G-1C-PC, MikroTik CCR1016-12G, MikroTik CCR1016-12S-1S+, MikroTik CCR1036-12G-4S-EM, MikroTik CCR1036-12G-4S, MikroTik CCR1036-8G-2S+, MikroTik CCR1036-8G-2S+EM , MikroTik CCR1072-1G-8S+, MikroTik CCR2004-16G-2S+, MikroTik CCR2004-1G-12S+2XS|RouterOS|MIKROTIK-MIB,HOST-RESOURCES-MIB|Certified| |MikroTik CRS<modelo de dispositivo> SNMP|MikroTik|MikroTik Cloud Router Switches (série CRS)|Modelos dedicados separados estão disponíveis para MikroTik CRS106-1C-5S, MikroTik CRS109-8G-1S-2HnD-IN, MikroTik CRS112-8G -4S-IN, MikroTik CRS112-8P-4S-IN, MikroTik CRS125-24G-1S-2HnD-IN, MikroTik CRS212-1G-10S-1S+IN, MikroTik CRS305-1G-4S+IN, MikroTik CRS309-1G- 8S+IN, MikroTik CRS312-4C+8XG-RM, MikroTik CRS317-1G-16S+RM, MikroTik CRS326-24G-2S+IN, MikroTik CRS326-24G-2S+RM, MikroTik CRS326-24S+2Q+RM, MikroTik CRS328-24P-4S+RM, MikroTik CRS328-4C-20S-4S+RM, MikroTik CRS354-48G-4S+2Q+RM, MikroTik CRS354-48P-4S+2Q+RM|RouterOS/SwitchOS|MIKROTIK-MIB,HOST -RESOURCES-MIB|Certificado| |MikroTik CSS<modelo de dispositivo> SNMP|MikroTik|MikroTik Cloud Smart Switches (série CSS)|Modelos dedicados separados estão disponíveis para MikroTik CSS326-24G-2S+RM, MikroTik CSS610-8G-2S+IN|RouterOS|MIKROTIK- MIB,HOST-RESOURCES-MIB|Certificado| |MikroTik FiberBox SNMP|MikroTik|MikroTik FiberBox|MikroTik FiberBox|RouterOS|MIKROTIK-MIB,HOST-RESOURCES-MIB|Certified| |MikroTik hEX <modelo de dispositivo> SNMP|MikroTik|MikroTik hEX|Modelos dedicados separados estão disponíveis para MikroTik hEX, MikroTik hEX lite, MikroTik hEX PoE, MikroTik hEX PoE lite, MikroTik hEX S|RouterOS|MIKROTIK-MIB,HOST-RECURSOS-MIB|Certificado| |MikroTik netPower <modelo de dispositivo> SNMP|MikroTik|MikroTik netPower|Modelos dedicados separados estão disponíveis para MikroTik netPower 15FR, MikroTik netPower 16P SNMP, MikroTik netPower Lite 7R|RouterOS/SwitchOS, SwitchOS Lite|MIKROTIK-MIB,HOST-RESOURCES -MIB|Certificado| |MikroTik PowerBox <modelo de dispositivo> SNMP|MikroTik|MikroTik PowerBox|Modelos dedicados separados estão disponíveis para MikroTik PowerBox, MikroTik PowerBox Pro|RouterOS|MIKROTIK-MIB,HOST-RESOURCES-MIB|Certified| |MikroTik RB<modelo de dispositivo> SNMP|MikroTik|Roteadores da série MikroTik RB|Modelos dedicados separados estão disponíveis para MikroTik RB1100AHx4, MikroTik RB1100AHx4 Dude Edition, MikroTik RB2011iL-IN, MikroTik RB2011iL-RMi, MikroTik RB2011iLS-IN, MikroTik RB2011U IN, MikroTik RB2011UiAS-RM, MikroTik RB260GS, MikroTik RB3011UiAS-RM, MikroTik RB4011iGS+RM, MikroTik RB5009UG+S+IN|RouterOS|MIKROTIK-MIB,HOST-RESOURCES-MIB|Certified| |MikroTik SNMP|MikroTik|MikroTik RouterOS devices|MikroTik CCR1016-12G, MikroTik RB2011UAS-2HnD, MikroTik 912UAG-5HPnD, MikroTik 941-2nD, MikroTik 951G-2HnD, MikroTik-1100AHx2|RouterOS|MIKRO MIB|Certificado| |QTech QSW SNMP|QTech|Dispositivos Qtech|Qtech QSW-2800-28T|QTECH-MIB,ENTITY-MIB|Inventário de desempenho| |Ubiquiti AirOS SNMP|Ubiquiti|Dispositivos sem fio Ubiquiti AirOS|NanoBridge,NanoStation,Unifi|AirOS|FROGFOOT-RESOURCES-MIB,IEEE802dot11-MIB|Desempenho| |HP Comware HH3C SNMP|HP|HP (H3C) Comware|HP A5500-24G-4SFP HI Switch|<|HH3C-ENTITY-EXT-MIB,ENTITY-MIB|Certified| |HP Enterprise Switch SNMP|HP|HP Enterprise Switch|HP ProCurve J4900B Switch 2626, HP J9728A 2920-48G Switch|<|STATISTICS-MIB,NETSWITCH-MIB,HP-ICF-CHASSIS,ENTITY-MIB,SEMI-MIB |Certificado| |TP-LINK SNMP|TP-LINK|TP-LINK|T2600G-28TS v2.0|<|TPLINK-SYSMONITOR-MIB,TPLINK-SYSINFO-MIB|Inventário de desempenho| |Netgear Fastpath SNMP|Netgear|Netgear Fastpath|M5300-28G|<|FASTPATH-SWITCHING-MIB,FASTPATH-BOXSERVICES-PRIVATE-MIB|Fault Inventory|

Design de modelo

Os modelos foram projetados com o seguinte em mente:

- As macros do usuário são usadas o máximo possível para que os gatilhos possam ser ajustados por o usuário;
- A descoberta de baixo nível é usada tanto quanto possível para minimizar o número de itens não suportados;
- Todos os modelos dependem do Template ICMP Ping para que todos os dispositivos também sejam verificado pelo ICMP;
- Os itens não usam MIBs - Os OIDs SNMP são usados em itens e de baixo nível descobertas. Portanto, não é necessário carregar nenhum MIB no Zabbix para modelos para trabalhar;
- As interfaces de rede de loopback são filtradas ao descobrir, bem como interfaces com ifAdminStatus = down(2)
- Contadores de 64 bits são usados de IF-MIB::ifXTable sempre que possível. Se isso não é suportado, os contadores padrão de 32 bits são usados;
- Todas as interfaces de rede descobertas têm um gatilho que controla sua status operacional (link).
 - Se você não quiser monitorar esta condição para um interface crie uma macro de usuário com contexto com o valor 0. Para exemplo:

Macro	Value
{#IFNAME}	0

onde Gi0/0 é {#IFNAME}. Dessa forma, o gatilho não é mais usado para esta interface específica.

* Você também pode alterar o comportamento padrão de todos os gatilhos para não disparar e ativar esse g

Macro	Value
{SIFCONTROL}	0
{SIFCONTROL: "Gi0/0"}	1
{SIFCONTROL: "Gi0/1"}	1

Tag

- Desempenho - os MIBs da família de dispositivos fornecem uma maneira de monitorar a CPU e itens de memória;
- Falha - os MIBs da família de dispositivos fornecem uma maneira de monitorar pelo menos um sensor de temperatura;
- Inventário - os MIBs da família de dispositivos fornecem uma maneira de coletar pelo menos o número de série do dispositivo e nome do modelo;
- Certificado - todas as três categorias principais acima são cobertas.

operação de modelo HTTP

Etapas para garantir a operação correta de modelos que coletam métricas com **agente HTTP**:

1. Crie um host no Zabbix e especifique um endereço IP ou nome DNS do alvo de monitoramento como a interface principal. Isso é necessário para o macro {HOST.CONN} para resolver corretamente nos itens do modelo.
2. [Link](#) o modelo para o host criado no passo 1 (se o modelo não estiver disponível na instalação do Zabbix, pode ser necessário importar o .xml do template arquivo primeiro - veja [Modelos pronto para uso](#) para instruções).
3. Ajuste os valores das macros obrigatórias conforme necessário.
4. Configure a instância que está sendo monitorada para permitir o compartilhamento de dados com Zabbix - veja as instruções na coluna Passos/comentários adicionais.

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo Readme.md do modelo (acessível por clicando em um nome de modelo).

[Modelo]Macros obrigatórios[Passos/comentários adicionais] |-----|-----|-----| |[Apache por HTTP](#)|**{\$APACHE.STATUS.H**
- o nome do host ou Endereço IP da página de status do Apache (padrão: 127.0.0.1).
{\$APACHE.STATUS.PATH} - o
caminho do URL (padrão: server-status?auto).
{\$ APACHE.STATUS.PORT} - a porta da página de status do Apache
(padrão: 80).
{\$APACHE.STATUS.SCHEME} - o esquema de solicitação. Suportado: http (padrão), https.|O mó-
dulo Apache mod_status deve ser definido (consulte Apache [documentação](#) para obter detalhes).
Para verificar a
disponibilidade, execute:
httpd -M 2>/dev/null \ | grep status_module

Exemplo de configuração do
Apache:
<Location "/server-status">
SetHandler server-status
Require host example.com
</Local>
[|Asterisco por HTTP](#)|**{\$AMI.PORT}** - Número da porta AMI para verificação disponibilidade do serviço (padrão: 8088).
{\$AMI.SECRET}
- o segredo do Asterisk Manager (padrão: zabbix).
{\$AMI.URL} - a API do Asterisk Manager URL no formato
<scheme>://<host>:<

http://asterisk:8088/asterisk/rawman).
 { \$AMI.USERNAME } - o nome do Asterisk Manager.|1. Ative o [servidor mini-HTTP](#).

2. Adicione a opção `webenabled=yes` à seção geral do arquivo `manager.conf`.

3. Crie um usuário Asterisk Manager na instância do Asterisk.| [ClickHouse por HTTP](#) **{ \$CLICKHOUSE.PORT }** - o porta do endpoint HTTP ClickHouse (padrão: 8123).
 { \$CLICKHOUSE.SCHEME } - o esquema de solicitação. Suportado: http (padrão), https.
 { \$CLICKHOUSE.USER }, { \$CLICKHOUSE.PASSWORD } - Credenciais de login do ClickHouse (nome de usuário padrão: zabbix, senha: zabbix_pass).
Se você não precisar de autenticação, remova os cabeçalhos dos itens do tipo de agente HTTP.|Crie um usuário ClickHouse com um perfil 'web' e permissão para visualizar bancos de dados (consulte [ClickHouse documentação](#) para obter detalhes).

Consulte o arquivo `Readme.md` do modelo para obter uma configuração de arquivo `zabbix.xml` pronta para uso.| [Cloudflare por HTTP](#) **{ \$CLOUDFLARE.API.TOKEN }** - Valor do token da API Cloudflare (padrão: '<change>').
 { \$CLOUDFLARE.ZONE_ID } - ID da zona do site Cloudflare (padrão: '<change>').| Os tokens de API da Cloudflare estão disponíveis na conta da Cloudflare em Meu perfil → Tokens de API.
O ID da zona está disponível na conta da Cloudflare em Início da conta → Site.| [DELL PowerEdge R720 por HTTP](#),
[DELL PowerEdge R740 por HTTP](#),
[DELL PowerEdge R820 por HTTP](#),
[DELL PowerEdge R840 por HTTP](#) **{ \$API.URL }** - URL da API Dell iDRAC Redfish no formato `<scheme>://<host>:<port>` (padrão: <Coloque seu URL aqui>)
 { \$API.USER }, { \$API.PASSWORD } - credenciais de login do Dell iDRAC (padrão: não definido).|Na interface Dell iDRAC do seu servidor:
1. Ative a API Redfish.
2. Crie um usuário para monitoramento com permissões somente leitura.| [Agrupamento Elasticsearch por HT](#)

operação de modelo ODBC

Etapas para garantir a operação correta de modelos que coletam métricas por meio de **Monitoramento ODBC**:

1. Certifique-se de que o driver ODBC necessário esteja instalado no servidor Zabbix ou procuração.
2. [Link](#) o modelo para um host de destino (se o modelo não estiver disponível em seu Instalação do Zabbix, pode ser necessário importar o arquivo .xml do template primeiro - veja [Modelos pronto para uso](#) para instruções).
3. Ajuste os valores das macros obrigatórias conforme necessário.
4. Configure a instância que está sendo monitorada para permitir o compartilhamento de dados com Zabbix - veja as instruções na coluna Passos/comentários adicionais.

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo `Readme.md` do modelo (acessível por clicando em um nome de modelo).

|Modelo|Macros obrigatórios|Passos/comentários adicionais| |-----|-----|-----| | [MSSQL por ODBC](#) **{ \$MSSQL.DSN }** - o nome da fonte de dados do sistema (padrão: <Coloque seu DSN aqui>)
 { \$MSSQL.PORT } - a porta TCP do Microsoft SQL Server (padrão: 1433)
 { \$MSSQL.USER }, { \$MSSQL.PASSWORD } - Credenciais de login do Microsoft SQL (padrão: não definido)|Crie um usuário do Microsoft SQL para monitoramento e conceda ao usuário as seguintes permissões: Exibir estado do servidor; Exibir qualquer definição (consulte [Microsoft SQL documentação](#) para obter detalhes).

O item "Estado da porta TCP do serviço" usa as macros `{HOST.CONN}` e `{ $MSSQL.PORT }` para verificar a disponibilidade da instância do Microsoft SQL.| [MySQL por ODBC](#) **{ \$MYSQL.DSN }** - o nome da fonte de dados do sistema (padrão: <Coloque seu DSN aqui>)
 { \$MYSQL.USER }, { \$MYSQL.PASSWORD } - credenciais de login do MySQL; a senha pode estar em branco (padrão: não definido)|Para conceder os privilégios necessários ao usuário MySQL que será usado para monitoramento, execute:
`GRANT USAGE, REPLICATION CLIENT, PROCESS, SHOW DATABASES, SHOW VIEW ON %% *.* TO ' <username> '@'%' ;%%

Veja documentação MySQL para detalhes .| Oracle by ODBC { $ORACLE.DSN } - o nome da fonte de dados do sistema (padrão: <Coloque seu DSN aqui>)
 { $ORACLE.PORT } - a porta TCP do banco de dados Oracle (padrão: 1521)
 { $ORACLE.USER }, { $ORACLE.PASSWORD } - Credenciais de login do Oracle (padrão: não definido)|1. Para criar um usuário Oracle para monitoramento, execute:
CREATE USER zabbix_mon IDENTIFIED BY <PASSWORD>;
-- Conceder acesso ao usuário zabbix_mon.
GRANT CONNECT, CREATE SESSION TO zabbix_mon;
GRANT SELECT ON V_$instance TO zabbix_mon;
GRANT SELECT ON V_$database TO zabbix_mon;
GRANT SELECT ON v_$sysmetric TO zabbix_mon;
GRANT SELECT ON v_$recovery_file_dest TO zabbix_mon;
GRANT SELECT ON v_$active_session_history TO zabbix_mon;
GRANT SELECT ON v_$osstat TO zabbix_mon;
GRANT SELECT ON v_$restore_point TO zabbix_mon;
GRANT SELECT ON v_$process TO zabbix_mon;
GRANT SELECT ON v_$datafile TO zabbix_mon;
GRANT SELECT ON v_$pgstat TO zabbix_mon;
GRANT SELECT ON v_$sgstat TO zabbix_mon;
GRANT SELECT ON v$log TO zabbix_mon;
GRANT SELECT ON v_$archive_dest TO zabbix_mon;
GRANT SELECT ON v_$asm_diskgroup TO zabbix_mon;
GRANT SELECT ON sys.dba_data_files TO zabbix_mon;
GRANT SELECT ON DBA_TABLESPACES TO zabbix_mon;
GRANT SELECT ON DBA_TABLESPACE_USAGE_METRICS TO zabbix_mon;
GRANT SELECT ON DBA_USERS TO zabbix_mon;

2. Certifique-se de que o ODBC se conecte ao Oracle com o parâmetro de sessão NLS_NUMERIC_CHARACTERS=' , '

3. Adicione um novo registro ao odbc.ini:
 [$ORACLE.DSN]
Driver = Oracle 19 ODBC driver
ServerName = $ORACLE.DSN
DSN = $ORACLE .DSN

4. Verifique a conexão via isql:
isql $TNS_NAME $DB_USER $DB_PASSWORD

5. Configure o servidor Zabbix ou o proxy Zabbix para uso do Oracle ENV. Edite ou adicione um novo arquivo: /etc/sysconfig/zabbix-server, ou para o proxy: /etc/sysconfig/zabbix-proxy. Em seguida, adicione as seguintes linhas ao arquivo:
export ORACLE_HOME=/usr/lib/oracle/19.6/client64
export PATH=$PATH:$ORACLE_HOME/bin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib64:/usr/lib:$ORACLE_HOME/bin
export TNS_ADMIN=$ORACLE_HOME/network/admin

Reinicie o servidor ou proxy Zabbix.`

Operação de template do agente Zabbix 2

Etapas para garantir a operação correta de modelos que coletam métricas com **agente Zabbix 2**:

1. Certifique-se de que o agente 2 esteja instalado no host e que a versão instalada contém o plugin necessário. Em alguns casos, você pode precisar **atualização** o agente 2 primeiro.
2. **Link** o modelo para um host de destino (se o modelo não estiver disponível em sua instalação do Zabbix, pode ser necessário importar o arquivo de importação do template primeiro - veja **Modelos prontos para uso** para instruções).
3. Ajuste os valores das macros obrigatórias conforme necessário. Observe que essas macros podem ser usadas para substituir parâmetros de configuração.
4. Configure a instância que está sendo monitorada para permitir o compartilhamento de dados com Zabbix - veja as instruções na coluna Passos/comentários adicionais.

::: não importante Os templates do Zabbix Agent 2 funcionam em conjunto com os plug-ins. Enquanto a configuração básica pode ser feita simplesmente ajustando as macros do usuário, a personalização mais profunda pode ser alcançada **configurando o plugin** em si. Por exemplo, se um plugin suporta sessões nomeadas, é possível para monitorar várias entidades do mesmo tipo (por exemplo, MySQL1 e MySQL2) especificando sessão nomeada com próprio URI, nome de usuário e senha para cada entidade no arquivo de configuração. :::

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo `Readme.md` do modelo (acessível por clicando em um nome de modelo).

[Nome do modelo]Macros obrigatórias|Passos/comentários adicionais|-----|-----|-----|-----| **Ceph by Zabbix agent 2** | **{\$CEPH.API.KEY}** - a chave da API (padrão: `zabbix_pass`).
Obrigatório, se `{$CEPH.CONNSTRING}` for um URI.
Deve estar vazio, se `{$CEPH.CONNSTRING}` for um nome de sessão.
{\$CEPH.CONNSTRING} - cadeia de conexão; pode ser um nome de sessão ou um URI definido no seguinte formato: `<protocol>(host:port)`. Para URI, apenas o esquema HTTPS é suportado.
Exemplos: `Prod, https://localhost:8003` (padrão)
{\$CEPH.USER} - usuário a ser usado para monitoramento (padrão: `zabbix`).
Obrigatório, se `{$CEPH.CONNSTRING}` for um URI.
Deve estar vazio, se `{$CEPH.CONNSTRING}` for um nome de sessão.|Funciona com o plugin Ceph; sessões nomeadas são suportadas.

1. Configure o Ceph RESTful Module de acordo com a **documentação**.

2. Certifique-se de que um endpoint de API RESTful esteja disponível para conexão.| **Docker** | Funciona com o plugin Docker; sessões nomeadas não são suportadas.

Para definir o caminho para o endpoint da API do Docker, edite o parâmetro `Plugins.Docker.Endpoint` no agente 2 **arquivo de configuração** (padrão: `Plugins.Docker.Endpoint=unix:///var/run/docker.sock`).

Para testar a disponibilidade, execute:
`zabbix_get -s docker-host -k docker.info` | **Memcached** | **{\$MEMCACHED.CONN.URI}** - conexão string no formato URI; a porta é opcional; a senha não é usada.
Se não for definido, o valor padrão do plugin é usado: `tcp://localhost:11211`.
Exemplos: `tcp://127.0.0.1:11211`, `tcp://localhost`, `unix: /var/run/memcached.sock`.|Funciona com o plug-in Memcached; sessões nomeadas são suportadas.

Para testar a disponibilidade, execute:
`zabbix_get -s memcached-host -k memcached.ping` | **Cluster MongoDB pelo agente Zabbix 2** | **{\$MONGODB.CONNSTRING}** * * - string de conexão no formato URI; a senha não é usada (padrão: `tcp://localhost:27017`).
Pode ser um nome de sessão ou um URI definido no seguinte formato: `%% <protocol>(host:port)>%%`
Para URI apenas o esquema TCP é suportado.
Exemplos: `MongoDB1, tcp://172.16.0.10`
`{$MONGODB.USER}`, `{$MONGODB.PASSWORD}` * * - credenciais do MongoDB (padrão: `none`).
Se não estiver definido e `{$MONGODB.CONNSTRING}` for um URI, os parâmetros do arquivo de configuração serão usados.
Deve estar vazio, se `{$MONGODB.CONNSTRING}` for um nome de sessão.|Funciona com plug-in MongoDB; sessões nomeadas são suportadas.
Para obter instruções de configuração do MongoDB, consulte **plugins**.
Para testar a disponibilidade, execute:
`zabbix_get -s mongos.node -k 'mongodb.ping["{$MONGODB.CONNSTRING}","{$MONGODB.USER}","{$MONGODB.PASSWORD}"]` | **Nó MongoDB pelo agente Zabbix 2** | `**{$MONGODB.CONNSTRING}* *` - linha de conexão

Operação do modelo do agente Zabbix

Etapas para garantir a operação correta de modelos que coletam métricas com **agente Zabbix**:

1. Certifique-se de que o agente Zabbix esteja instalado no host. Para ativar verificações, também certifique-se de que o host seja adicionado ao 'ServerActive' parâmetro do agente **configuração arquivo**.
2. **Link** o modelo para um host de destino (se o modelo não estiver disponível em sua instalação do Zabbix, pode ser necessário importar o arquivo .xml do template primeiro - veja **Modelos prontos para uso** para instruções).
3. Ajuste os valores das macros obrigatórias conforme necessário.
4. Configure a instância que está sendo monitorada para permitir o compartilhamento de dados com Zabbix - veja as instruções na coluna Passos/comentários adicionais.

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo `Readme.md` do modelo (acessível por clicando em um nome de modelo).

[Nome do modelo]Macros obrigatórios|Passos/comentários adicionais|-----|-----|-----|-----| [Apache pelo agente Zabbix]{\$APACHE.STATUS.HOST}** - o nome do host ou endereço IP da página de status do Apache (padrão: 127.0.0.1)
{\$APACHE.STATUS.PATH}** - o caminho da URL (padrão: server-status?auto)
{\$APACHE.STATUS.PORT} - a porta da página de status do Apache (padrão: 80)|modulo do Apache mod_status deve ser definido (consulte Apache documentação para detalhes).
Para verificar a disponibilidade, execute:
httpd -M 2>/dev/null \|| grep status_module

Exemplo de configuração do Apache:
<Location "/server-status">
SetHandler server-status
Require host example.com
</Local>| [HAProxy pelo agente Zabbix]{\$HAPROXY.STATS.PATH} * - o caminho da página de estatísticas do HAProxy (padrão: stats)
{\$HAPROXY.STATS.PORT} - a porta do host ou contêiner do HAProxy Stats (padrão: 8404)
{\$HAPROXY .STATS.SCHEME} - o esquema da página HAProxy Stats. Com suporte: http (padrão), https|HAProxy Stats página deve ser configurada (consulte HAProxy post do blog para obter detalhes ou Readme.md do template para exemplo de configuração).| [IIS por agente Zabbix / IIS por agente Zabbix ativo]{\$IIS.PORT} - a porta que o servidor IIS escuta (padrão: 80)
* {\$IIS.SERVICE} - o serviço para verificação de porta (padrão: http). Consulte a seção net.tcp.service para obter detalhes.|O servidor deve ter as seguintes funções:
Web Server
Scripts e ferramentas de gerenciamento do IIS

Consulte a documentação do IIS para obter detalhes.| [Microsoft Exchange Server 2016 por agente Zabbix/Microsoft Exchange Server 2016 por agente Zabbix ativo]
|Observe que o modelo não fornece informações sobre o estado dos serviços do Windows . Recomenda-se usá-lo junto com o modelo OS Windows pelo agente Zabbix ou OS Windows pelo agente Zabbix ativo.| [Nginx por agente Zabbix]{\$NGINX.STUB_STATUS.HOST} - o nome do host ou endereço IP do host ou contêiner Nginx stub_status (padrão: localhost)
{\$NGINX.STUB_STATUS.PATH} - o caminho da página Nginx stub_status (padrão : basic_status)
{\$NGINX.STUB_STATUS.PORT} - a porta do host ou contêiner Nginx stub_status (padrão: 80)|ngx_http_stub_status_module deve ser configurada (consulte Nginx documentação para obter detalhes ou o Readme.md do modelo para exemplo de configuração).
Para verificar a disponibilidade, execute:
nginx -V 2 >&1 \|| grep -o with-http_stub_status_module| [PHP-FPM pelo agente Zabbix]{\$PHP_FPM .HOST} - um nome de host ou um IP do host ou contêiner de status do PHP-FPM (padrão: localhost)
{\$PHP_FPM.PING.PAGE} - caminho da página de ping do PHP-FPM (padrão:ping)
{\$PHP_FPM.PORT}** - a porta

operação do modelo IPMI

Os modelos IPMI não requerem nenhuma configuração específica. Para iniciar o monitoramento, link o modelo para um host de destino (se o template não estiver disponível em seu Zabbix instalação, pode ser necessário importar primeiro o arquivo .xml do modelo - veja Modelos prontos para uso seção para instruções).

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo Readme.md do modelo (acessível por clicando em um nome de modelo).

[Modelo]Macros obrigatórios|Passos/comentários adicionais|-----|-----|-----|-----| [Chassis by IPMI]{\$IPMI.USER}, * {\$IPMI.PASSWORD}** - credenciais para acesso ao BMC (padrão: nenhum)|-

Operação do modelo JMX

Etapas para garantir a operação correta de modelos que coletam métricas por JMX:

1. Certifique-se de que o Zabbix Java gateway está instalado e configurar corretamente.
2. Link o modelo para o host de destino. O host deve ter a interface JMX configurada.
Se o template não estiver disponível em sua instalação do Zabbix, você pode precisa importar primeiro o arquivo .xml do modelo - veja Modelos pronto para uso para instruções.
3. Ajuste os valores das macros obrigatórias conforme necessário.
4. Configure a instância que está sendo monitorada para permitir o compartilhamento de dados com Zabbix - veja as instruções na coluna Passos/comentários adicionais.

Esta página contém apenas um conjunto mínimo de macros e etapas de configuração necessárias para a operação adequada do modelo. Uma detalhada descrição de um modelo, incluindo a lista completa de macros, itens e triggers, está disponível no arquivo Readme.md do modelo (acessível por clicando em um nome de modelo).

[Modelo]Macros obrigatórios|Passos/comentários adicionais|-----|-----|-----|-----| [Apache ActiveMQ por JMX]{\$ACTIVEMQ.PORT} - porta para JMX (padrão: 1099).
{\$ACTIVEMQ.USERNAME}, {\$ACTIVEMQ.PASSWORD} - credenciais de login para JMX
(nome de usuário padrão: admin, senha: activemq).|O acesso JMX ao Apache ActiveMQ deve ser habilitado e configurado de acordo com as instruções na documentação oficial.| [Apache Cassandra por JMX]{\$CASSANDRA.USER}, {\$CASSANDRA.PASSWORD} - Credenciais de login do Apache Cassandra
(nome de usuário padrão: zabbix, senha: zabbix)|O acesso JMX ao Apache Cassandra deve ser habilitado e configurado de acordo com as instruções na documentação oficial.| [Apache Kafka por JMX]{\$KAFKA.USER}, {\$KAFKA.PASSWORD} - Credenciais de login do Apache Kafka
(nome de usuário padrão: zabbix, senha: zabbix)|O acesso JMX ao Apache Kafka deve ser habilitado e configurado de acordo com as instruções

na [documentação oficial](#).| [Apache Tomcat por JMX](#)|`{${TOMCAT.USER}}, {${TOMCAT.PASSWORD}}` - Credenciais de login do Apache Tomcat; deixe em branco se a instalação do Tomcat não exigir autenticação
(padrão: não definido).|O acesso JMX ao Apache Tomcat deve ser habilitado e configurado de acordo com as instruções na [documentação oficial](#) (escolha a versão correta).| [GridGain by JMX](#)|`{${GRIDGAIN.USER}}, * {${GRIDGAIN.PASSWORD}}` - Credenciais de login do GridGain (nome de usuário padrão: zabbix, senha: <secret>).|O acesso JMX à plataforma de computação em memória GridGain deve ser ativado e configurado de acordo com as instruções na [documentação](#).| [Ignite by JMX](#)|`{${IGNITE.USER}}, * {${IGNITE.PASSWORD}}` - Credenciais de login do Apache Ignite (nome de usuário padrão: zabbix, senha: <secret>).|Ative e configure o acesso JMX ao Apache Ignite.

A hierarquia da árvore JMX contém ClassLoader por padrão . Adicionar a seguinte opção de máquina virtual Java `-DIGNITE_MBEAN_APPEND_CLASS_LOADER_ID=false` excluirá um nível com o nome do ClassLoader.

As métricas de cache e região de dados podem ser configuradas conforme necessário - consulte [documentação do Ignite](#) para obter detalhes.

Veja também: [Monitoring and Management Using JMX Technology](#)| [WildFly Domain by JMX](#)|`{${WILDFLY.JMX.PROTOCOL}} *` - Esquema JMX (padrão: remote+http)
|`{${WILDFLY.USER}}, {${WILDFLY.PASSWORD}}` - Credenciais de login do WildFly (nome de usuário padrão: zabbix, senha: zabbix). |1. Habilite e configure o acesso JMX ao WildFly de acordo com as instruções na [documentação oficial](#).

2. Copie jboss-client.jar de `/(wildfly,EAP,Jboss,AS)/bin/client` para o diretório `/usr/share/zabbix-java-gateway/lib`.

3. Reinicie o Zabbix Java Gateway.| [WildFly Server by JMX](#)|`{${WILDFLY.JMX.PROTOCOL}} *` - Esquema JMX (padrão: remote+http)
|`{${WILDFLY.USER}}, {${WILDFLY.PASSWORD}}` - Credenciais de login do WildFly (nome de usuário padrão: zabbix, senha: zabbix). |1. Habilite e configure o acesso JMX ao WildFly de acordo com as instruções na [documentação oficial](#).

2. Copie jboss-client.jar de `/(wildfly,EAP,Jboss,AS)/bin/client` para o diretório `/usr/share/zabbix-java-gateway/lib`.

3. Reinicie o Zabbix Java Gateway.

10 Notificações sobre eventos

Visão geral

Assumindo que tenhamos configurado alguns itens e gatilhos e agora temos alguns eventos ocorrendo como resultado de mudanças de estado de gatilhos, é o momento de considerar algumas ações.

Para começar, não gostaríamos de consultar a lista de gatilhos e eventos o tempo todo. Seria muito melhor receber notificação se algo significativo (como um problema) ocorresse. Também, quando problemas ocorrem, gostaríamos de ver que todas as pessoas interessadas foram informadas.

É por isso que o envio de notificações é uma das principais ações oferecidas pelo Zabbix. Quem e quando deve ser notificado sobre um certo evento pode ser definido.

Para estar apto a enviar e receber notificações do Zabbix você tem que:

- **definir alguma mídia**
- **configurar uma ação** que envia uma mensagem para uma das mídias definidas

Ações consistem de condições e operações. Basicamente, quando condições são atendidas, operações são realizadas. As duas principais operações são enviar uma mensagem (notificação) e executar um comando remoto.

Para eventos criados por descoberta e autoregistro, algumas operações adicionais estão disponíveis. Essas incluem adicionar e remover um host, associar um template, etc.

1 Tipos de mídia

Visão geral

Mídias são os canais de entrega usados para envio de notificações e alertas do Zabbix.

Você pode configurar vários tipos de mídia:

- **E-mail**
- **SMS**
- **Scripts de alerta customizados**
- **Webhook**

Tipos de mídia são configurados em Administração → Tipos de mídia.

<input type="checkbox"/>	Name ▲	Type	Status	Used in actions	Details	Action
<input type="checkbox"/>	E-mail	Email	Enabled	Report problems to Zabbix administrators	SMTP server: "mail.example.com", SMTP helo: "example.com", SMTP email: "zabbix@example.com"	Test
<input type="checkbox"/>	Jira	Webhook	Enabled			Test
<input type="checkbox"/>	Mattermost	Webhook	Enabled			Test
<input type="checkbox"/>	Opsgenie	Webhook	Enabled			Test
<input type="checkbox"/>	Pushover	Webhook	Enabled			Test
<input type="checkbox"/>	Slack	Webhook	Enabled			Test
<input type="checkbox"/>	SMS	SMS	Enabled		GSM modem: "/dev/ttyS0"	Test
<input type="checkbox"/>	Test THROW	Webhook	Enabled			Test
<input type="checkbox"/>	Zendesk	Webhook	Enabled			Test

Alguns tipos de mídia vêm pré-definidos no conjunto de dados padrão. Você precisa apenas ajustar seus parâmetros para tê-los funcionando.

É possível testar se um tipo de mídia configurado funciona, clicando em Testar na última coluna (consulte teste de tipos de mídia para mais detalhes).

Para criar um novo tipo de mídia, clique no botão Criar tipo de mídia. Um formulário de configuração de tipo de mídia é aberto.

Parâmetros comuns

Alguns parâmetros são comuns para todos os tipos de mídia.

Media type

Message templates 5

Options

*

Name

SMS

Type

SMS

▼

*

GSM modem

/dev/ttyS0

Description

Enabled

☒

Add

Cancel

Na aba **Tipo de mídia** os atributos gerais comuns são:

Parâmetro	Descrição
Nome	Nome do tipo de mídia.
Tipo	Selecione o tipo de mídia.
Descrição	Informe uma descrição.
Habilitada	Marque esta caixa para habilitar o tipo de mídia.

Veja as páginas individuais de tipos de mídia para parâmetros de mídia específicos.

A aba **Modelos de mensagem** permite configurar as mensagens de notificação padrão para todos ou alguns dos seguintes tipos de eventos:

- Problema
- Recuperação de problema
- Atualização de problema
- Serviço
- Recuperação de serviço
- Atualização de serviço
- Descoberta
- Autoregistro
- Problema interno
- Recuperação de problema interno

Media types

[Media type](#)
[Message templates 6](#)
[Options](#)

Message type	Template	Actions
Problem	Problem started at {EVENT.TIME} on {EVENT.DA...}	Edit Remove
Problem recovery	Problem has been resolved at {EVENT.RECOVE...}	Edit Remove
Problem update	{USER.FULLNAME} {EVENT.UPDATE.ACTION} prob...	Edit Remove
Service	Service problem started at {EVENT.TIME} on {EV...}	Edit Remove
Service recovery	Service "{SERVICE.NAME}" has been resolved a...	Edit Remove
Autoregistration	Host name: {HOST.HOST} Host IP: {...	Edit Remove

[Add](#)

Para customizar os modelos de mensagem:

- Na aba Modelos de mensagem clique em [Add](#) uma janela de Modelo de mensagem será aberta.
- Selecione o Tipo de mensagem requerido e edite os textos de Assunto e Mensagem.
- Clique em Adicionar para salvar o modelo de mensagem

Message template

Message type

Problem

Subject

Problem: {EVENT.NAME}

Message

Problem started at {EVENT.TIME} on {EVENT.DATE}
Problem name: {EVENT.NAME}
Host: {HOST.NAME}
Severity: {EVENT.SEVERITY}
Operational data: {EVENT.OPDATA}
Original problem ID: {EVENT.ID}
{TRIGGER.URL}

Add

Cancel

Parâmetros de modelo de mensagem:

Parâmetro	Descrição
Tipo de mensagem	Tipo de um evento para o qual a mensagem padrão deve ser usada. Apenas uma mensagem padrão pode ser definida para cada tipo de evento.
Assunto	Assunto da mensagem padrão. O assunto pode conter macros. É limitado a 255 caracteres. O assunto não está disponível para tipo de mídia SMS.
Mensagem	A mensagem padrão. É limitado a uma certa quantidade de caracteres dependendo do tipo de banco de dados (consulte Envio de mensagens para mais informações). A mensagem pode conter macros suportadas. Em mensagens de problema e atualização de problema, macros de expressão são suportadas (por exemplo, {?avg(/host/key,1h)}).

Para fazer alterações em um modelo de mensagem existente: Na coluna Ações clique em [Edit](#) para editar o modelo ou clique em [Remove](#) para eliminar o modelo de mensagem.

É possível definir um modelo de mensagem customizado para uma ação específica (veja [operações de ação](#) para detalhes). Mensagens customizadas definidas na configuração de ação irão sobrescrever o modelo de mensagem padrão do tipo de mídia.

Warning:

Definição de modelos de mensagem é obrigatório para todos os tipos de mídia, incluindo webhooks ou scripts de alerta customizados que não usam mensagens padrão para notificações. Por exemplo, uma ação "Enviar mensagem para Pushover webhook" falhará em enviar notificações de problema, se a mensagem de Problema para o Pushover webhook não estiver definida.

A tabela **Opções** contém configurações de processamento de alerta. O mesmo conjunto de opções é configurável para cada tipo de mídia.

Todos os tipos de mídia são processados em paralelo. Enquanto o número máximo de sessões concorrentes é configurável por tipo de mídia, o número total de processos de alerta no Server pode ser limitado apenas pelo [parâmetro](#) StartAlerters. Alertas gerados por um gatilho são processados sequencialmente. Então múltiplas notificações podem ser processadas simultaneamente apenas se elas forem geradas por múltiplos gatilhos.

Concurrent sessions

One

Unlimited

Custom

* Attempts

3

* Attempt interval

10s

Parâmetro	Descrição
Sessões concorrentes	<p>Selecione o número de sessões de alerta paralelas para o tipo de mídia:</p> <p>One - uma sessão</p> <p>Unlimited - número de sessões ilimitadas</p> <p>Custom - selecione um número de sessões customizados</p> <p>Valores ilimitados/altos significam mais sessões paralelas e capacidade aumentada para envio de notificações. Valores ilimitados/altos devem ser usados em grandes ambientes onde o envio simultâneo de muitas notificações possa ser necessário. Se mais notificações precisam ser enviadas do que sessões concorrentes existentes, as notificações restantes serão enfileiradas; elas não serão perdidas.</p>
Tentativas	<p>Número de tentativas de envio de uma notificação. Até 100 tentativas podem ser especificadas; o valor padrão é '3'. Se '1' estiver especificado o Zabbix enviará a notificação apenas uma vez e não fará uma nova tentativa caso o envio falhe.</p>
Intervalo de tentativa	<p>Frequência de tentativas de reenviar uma notificação no caso de falha de envio, em segundos (0-3600). Se '0' for especificado, o Zabbix fará uma nova tentativa imediatamente.</p> <p>Sufixos de tempo são suportados, p.e. 5s, 3m, 1h.</p>

Teste de tipo de mídia

É possível testar se um tipo de mídia funciona.

E-mail

Por exemplo, para testar um tipo de mídia e-mail:

- Localize o e-mail relevante na [lista](#) de tipos de mídia
- Clique em Testar na última coluna da lista (uma janela de teste será aberta)
- Informe um endereço em Enviar para com corpo e assunto opcional
- Envie uma mensagem de texto clicando em Testar

A mensagem de sucesso ou falha da mensagem será exibida na mesma janela:

Test media type

Media type test successful.

* Send to

address@domain.com

Subject

Test subject

* Message

This is the test message from Zabbix

Test

****Webhook ****

Para testar um tipo de mídia webhook:

- Localize o webhook relevante na **lista** de tipos de mídia
- Clique em Testar na última coluna da lista (uma janela de teste será aberta)
- Edite os valores de parâmetro de webhook, se necessário
- Clique em Testar

Por padrão, testes de webhook são executados com parâmetros informados durante a configuração. No entanto, é possível alterar os valores de atributo para teste. Substituição ou eliminação de valores na janela de teste afeta apenas o procedimento de teste, os valores de atributo atuais permanecerão inalterados.

Test media type "Telegram webhook"

Media type test successful.

Message
{ALERT.MESSAGE}

telegramTOKEN
1266457374:AAFqF072oyxROyWyAGU9hsf_vqcXvYVmyxl

To
{ALERT.SENDTO}

URL
{\$Zabbix_URL}

Response

```
{
  "tags": {
    "key": "MSG-115",
    "link": "http://example.com/MSG-115"
  }
}
```

Response type: JSON
[Open log](#)

Test
Cancel

Para visualizar as entradas de log do teste de tipo de mídia sem deixar a janela de teste:

- Clique em Abrir log (uma nova janela será aberta).

Test media type "Telegram"

Details
Media type test failed.

Sending failed: Bad Request: chat not found.

Name
Type

Email
Email

Mattermost
Webhook

Message
{ALERT.MESSAGE}

Media type test log

```
00:00:00.000 [Debug] [Telegram Webhook] URL: https://api.telegram.org/bot<TOKEN>/sendMessage
00:00:00.000 [Debug] [Telegram Webhook] params: {"chat_id":"{ALERT.SENDTO}","text":"{ALERT.SUBJECT}\n{ALERT.MESSAGE}","disable_web_page_preview":true}
00:00:00.139 [Debug] [Telegram Webhook] HTTP code: 400
00:00:00.140 [Debug] [Telegram Webhook] notification failed: Bad Request: chat not found

Time elapsed: 140ms
```

Ok

selected
Enable
Disable

Response type: String
[Open log](#)

Test
Cancel

Se o teste de webhook for bem-sucedido

- Uma mensagem "Teste de tipo de mídia bem-sucedido." é exibida
- A resposta do Server aparece no campo cinza Resposta
- Tipo de resposta (JSON ou String) é especificado abaixo do campo Resposta

Se o teste de webhook falhar

- Uma mensagem "Teste de tipo de mídia falhou." é exibida, seguida de detalhes adicionais sobre a falha.

Mídia de usuário

Para receber notificações de um tipo de mídia, um meio (endereço de e-mail/ número de telefone/ID de usuário webhook/etc.) para este tipo de mídia deve ser definido no perfil do usuário. Por exemplo, uma ação de envio de mensagens para o usuário "Admin" usando o webhook "X" sempre falhará em enviar qualquer coisa se o meio webhook "X" não estiver definido no perfil do usuário.

Para definir mídia de usuário:

- Vá até seu perfil de usuário, ou vá até Administração → Usuários e abra o formulário de propriedades do usuário
- Na aba de Mídia, clique em [Add](#)

Media

Type

Email

*

Send to

example@company.com

Remove

example recipient <example2@company.com>

Remove

Add

*

When active

1-7,00:00-24:00

Use if severity

☒ Not classified

☒ Information

☒ Warning

☒ Average

☒ High

☒ Disaster

Enabled

☒

Update

Cancel

Atributos de mídia de usuário:

Parâmetro	Descrição
Tipo	A lista contém os nomes de todos os tipos de mídia configurados.
Enviar para	Forneça as informações de contato necessárias para onde enviar as mensagens. Para um tipo de mídia e-mail é possível adicionar vários endereços clicando em Add abaixo do campo de endereço. Neste caso, a notificação será enviada para todos os endereços de e-mail fornecidos. É também possível especificar o nome de destinatário no campo Enviar para do destinatário de e-mail no formato 'Nome de destinatário <endereço1@company.com>'. Note que se um nome de destinatário é fornecido, um endereço de e-mail deve ser envolto em colchetes angulares (<>). Caracteres UTF-8 são suportados no nome, pares quotados e comentários não são. Por exemplo: John Abercroft <gerente@nycdatcenter.com> e gerente@nycdatcenter.com são ambos formatos válidos. Exemplos incorretos: John Doe zabbix@company.com, %%"Zabbix\@<H(comment)Q\>" zabbix@company.com %%.

Parâmetro	Descrição
Quando ativo	Você pode limitar o horário de envio das mensagens, por exemplo, configure para dias úteis somente (1-5,09:00-18:00). Note que este limite é baseado na time zone do usuário. Se a time zone for alterada e for diferente da time zone do sistema este limite pode precisar ser ajustado de acordo, para que mensagens importantes não sejam perdidas. Veja a página de especificação de período de tempo para descrição do formato.
Use se severidade	Marque as caixas de severidades de gatilho para as quais você quer receber notificações. Note que a severidade padrão ('Não classificado') deve ser marcada se você quiser receber notificações para eventos sem gatilho. Após salvar, as severidades de gatilho selecionadas serão exibidas nas cores de severidade correspondentes, enquanto as não selecionadas ficarão cinzas.
Estado	Estado da mídia do usuário. Habilitada - está em uso. Desabilitada - não está sendo usada.

1 E-mail

Visão Geral

Para configurar o e-mail como o canal de entrega para mensagens, você precisa configurar o e-mail como o tipo de mídia e atribuir endereços específicos aos usuários.

Note:

Múltiplas notificações para um único evento serão agrupadas juntas na mesma thread de e-mail.

Configuração

Para configurar o e-mail como tipo de mídia:

- Acesse Administração → Tipos de mídia
- Clique em Criar tipo de mídia (ou clique em E-mail na lista de tipos de mídia predefinidos).

A aba de **Tipo de mídia** contém atributos gerais do tipo de mídia:

Media type
Message templates 5
Options

* Name

Email

Type

Email

* SMTP server

mail.example.com

SMTP server port

25

* SMTP helo

example.com

* SMTP email

Zabbix_info <zabbix@example.com>

Connection security

None

STARTTLS

SSL/TLS

Authentication

None

Username and password

Message format

HTML

Plain text

Description

Enabled

☒

Todos os campos de entrada obrigatórios são marcados com um asterisco vermelho.

Os seguintes parâmetros são específicos para o tipo de mídia de e-mail:

Parametro	Descrição
Servidor SMTP	Defina um servidor SMTP para lidar com mensagens de saída.
Porta do servidor SMTP	Defina a porta do servidor SMTP para lidar com mensagens de saída. Esta opção é suportada a partir do Zabbix 3.0.
SMTP helo	Defina um valor SMTP helo correto, normalmente um nome de domínio.

Parametro	Descrição
E-mail SMTP	<p>O endereço inserido aqui será usado como endereço De para as mensagens enviadas.</p> <p>A adição de um nome de exibição do remetente (como "Zabbix_info" em Zabbix_info <zabbix@company.com> na captura de tela acima) com o endereço de e-mail real é suportada desde a versão Zabbix 2.2.</p> <p>Existem algumas restrições nos nomes de exibição nos e-mails do Zabbix em comparação com o que é permitido pelo RFC 5322, como ilustrado pelos exemplos:</p> <p>Exemplos válidos:</p> <p>zabbix@company.com (somente endereço de e-mail, não é necessário usar colchetes angulares)</p> <p>Zabbix_info <zabbix@company.com> (exibir nome e endereço de e-mail em colchetes angulares)</p> <p>ΣΩ-monitoring <zabbix@company.com> (caracteres UTF-8 no nome de exibição)</p> <p>Exemplos inválidos:</p> <p>Zabbix HQ zabbix@company.com (nome de exibição presente, mas sem colchetes angulares no endereço de e-mail)</p> <p>"Zabbix\@<H(comment)Q\>" <zabbix@company.com> (embora válido pelo RFC 5322, pares entre aspas e comentários não são suportados em e-mails do Zabbix)</p>
Segurança da conexão	<p>Selecione o nível de segurança da conexão:</p> <p>Nenhum - não use a opção CURLOPT_USE_SSL</p> <p>STARTTLS - use a opção CURLOPT_USE_SSL com valor CURLUSESSL_ALL</p> <p>SSL/TLS - uso de CURLOPT_USE_SSL é opcional</p> <p>Esta opção é suportada a partir do Zabbix 3.0.</p>
Verificar par de SSL	<p>Marque a caixa de seleção para verificar o certificado SSL do servidor SMTP.</p> <p>O valor da diretiva de configuração do servidor "SSLCAlocation" deve ser colocado em CURLOPT_CAPATH para validação de certificado.</p> <p>Isto define a opção CURLOPT_SSL_VERIFYPEER.</p> <p>Esta opção é suportada a partir do Zabbix 3.0.</p>
Verificar host de SSL	<p>Marque a caixa de seleção para verificar se o campo Nome Comum ou o campo Nome Alternativo do Assunto do certificado do servidor SMTP corresponde.</p> <p>Isto define a opção CURLOPT_SSL_VERIFYHOST.</p> <p>Esta opção é suportada a partir do Zabbix 3.0.</p>
Autenticação	<p>Selecione o nível de autenticação:</p> <p>Nenhum - nenhuma opção cURL é definida</p> <p>(desde 3.4.2) Nome do usuário e senha - implica "AUTH=*" deixando a escolha do mecanismo de autenticação para cURL</p> <p>(até 3.4.2) Senha normal - CURLOPT_LOGIN_OPTIONS é definido como "AUTH=PLAIN"</p> <p>Esta opção é suportada a partir do Zabbix 3.0.</p>
Nome do usuário	<p>Nome do usuário a ser usado na autenticação.</p> <p>Isto define o valor de CURLOPT_USERNAME.</p> <p>Esta versão é suportada a partir do Zabbix 3.0.</p>
Senha	<p>Senha a ser usada na autenticação.</p> <p>Isto define o valor de CURLOPT_PASSWORD.</p> <p>Esta versão é suportada a partir do Zabbix 3.0.</p>
Formato da mensagem	<p>Selecione o formato da mensagem:</p> <p>HTML - enviar como HTML</p> <p>Texto simples - enviar texto simples</p>

Attention:

Para habilitar as opções de autenticação SMTP, o servidor Zabbix deve ser compilado com a opção de compilação `--with-libcurl compilation` (com cURL 7.20.0 ou superior) e usar os pacotes `libcurl-full` durante a execução.

Para detalhes sobre como configurar mensagens padrão e opções de processamento de alerta, consulte [common media type parameters](#)

Mídia de usuário

Uma vez que o tipo de mídia e-mail é configurado, vá até a seção Administração → Usuários e edite o perfil do usuário para associar o tipo de mídia e-mail para o usuário. As etapas para configuração de mídia de usuário, sendo comum para todos os tipos de mídia, estão descritas na página [Tipos de mídia](#).

2 SMS

Visão geral

O Zabbix suporta o envio de mensagens SMS usando um modem GSM serial conectado à porta serial do servidor do Zabbix.

Certifique-se que:

- A velocidade do dispositivo serial (normalmente /dev/ttyS0 no Linux) corresponde à do modem GSM. O Zabbix não define a velocidade do link serial. Ele usa as configurações padrão.
- O usuário 'zabbix' tem acesso de leitura/escrita ao dispositivo serial. Execute o comando `ls -l /dev/ttyS0` para ver as permissões atuais do dispositivo serial.
- O modem GSM tem o PIN informado e o preserva após reinício de energia. Alternativamente você pode desabilitar o PIN no cartão SIM. O PIN pode ser informado ao emitir o comando `AT+CPIN="NNNN"` (NNNN é seu número PIN, as aspas devem estar presentes) em um software de terminal, como o minicom do Unix ou HyperTerminal no Windows.

O Zabbix foi testado com estes modems GSM:

- Siemens MC35
- Teltonika ModemCOM/G10

Para configurar SMS como o canal de entrega para mensagens, você também precisa configurar o SMS como o tipo de mídia e informar os respectivos números de telefone para os usuários.

Configuração

Para configurar SMS como o tipo de mídia:

- Vá até Administração → Tipos de mídia
- Clique em Criar tipo de mídia (ou clique em SMS na lista de tipos de mídia pré-definidos).

Os seguintes parâmetros são específicos para o tipo de mídia SMS:

Parâmetro	Descrição
Modem GSM	Define o nome do dispositivo serial do modem GSM.

Consulte os [parâmetros de tipo de mídia comuns](#) para detalhes em como configurar opções de mensagens padrão e processamento de alerta. Note que o processamento paralelo de envio de notificações SMS não é possível.

Mídia de usuário

Uma vez que o tipo de mídia SMS é configurado, vá até a seção Administração → Usuários e edite o perfil do usuário para associar o tipo de mídia SMS para o usuário. As etapas para configuração de mídia de usuário, sendo comum para todos os tipos de mídia, são descritas na página [Tipos de mídia](#).

3 Scripts de alerta customizados

Visão geral

Se você não estiver satisfeito com os tipos de mídia existentes para o envio de alertas há um caminho alternativo. Você pode criar um script que lidará com a notificação do seu jeito.

Scripts de alerta são executados no Zabbix Server. Estes scripts estão localizados no diretório definido pela variável **AlertScriptsPath** no [arquivo de configuração](#) do Server.

Aqui temos um exemplo de script de alerta:

```
#####!/bin/bash

to=$1
subject=$2
body=$3

cat <<EOF | mail -s "$subject" "$to"
$body
EOF
```

Attention:

A partir da versão 3.4 o Zabbix verifica pelo código de saída dos comandos e scripts executados. Qualquer código de saída diferente de **0** é considerado como um erro de **execução de comando**. Neste caso o Zabbix tentará repetir a execução falha.

Variáveis de ambiente não são preservadas ou criadas para o script, então elas devem ser manipuladas de forma explícita.

Configuração

Para configurar scripts de alerta customizados como o tipo de mídia:

- Vá até Administração → Tipos de mídia
- Clique em Criar tipo de mídia

A aba **Tipo de mídia** contém atributos de tipo de mídia gerais:

Media type

Message templates

Options

*

Name

Notification script

Type

Script

*

Script name

notification.sh

Script parameters

Parameter

{ALERT.SENDTO}

{ALERT.SUBJECT}

{ALERT.MESSAGE}

Add

Description

Enabled

☒

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Os seguintes parâmetros são específicos para o tipo de mídia script:

Parâmetro	Descrição
Nome do script	Informe o nome do script.
Parâmetros do script	Adicione parâmetros de linha de comando ao script. As macros {ALERT.SENDTO}, {ALERT.SUBJECT} e {ALERT.MESSAGE} são suportadas nos parâmetros de script. Customização de parâmetros de script é suportada desde o Zabbix 3.0.

Consulte os **parâmetros de tipo de mídia comuns** para detalhes em como configurar opções de mensagens padrão e processamento de alerta.

Warning:

Mesmo que um script de alerta não use mensagens padrão, os modelos de mensagem para tipos de operação usados por este tipo de mídia ainda precisam ser definidos, caso contrário a notificação não será enviada.

Attention:

Como o processamento paralelo de tipos de mídia está implementada desde o Zabbix 3.4.0, é importante notar que com mais de um tipo de mídia script configurado, estes scripts podem ser processados em paralelo pelos processos de alerta. O número total de processos de alerta é limitado pelo **parâmetro** StartAlerters.

Mídia de usuário

Uma vez que o tipo de mídia é configurado, vá até a seção Administração → Usuários e edite o perfil do usuário associando a mídia deste tipo ao usuário. As etapas para configuração de mídia de usuário, sendo comum para todos os tipos de mídia, são descritas na página [Tipos de mídia](#).

Note que quando definindo a mídia de usuário, o campo Enviar para não pode estar vazio. Se este campo não será usado no script de alerta, entre qualquer combinação de caracteres suportados para atender aos requisitos de validação.

4 Webhook

Visão geral

O tipo de mídia webhook é útil para chamadas HTTP customizadas de código JavaScript para integração direta com software externos como sistemas de helpdesk, chats, ou mensageiros. Você pode escolher importar uma integração provida pelo Zabbix ou criar uma integração customizada do zero.

Integrações

As seguintes integrações estão disponíveis permitindo o uso de tipos de mídia webhook pré-definidos para envio de notificações Zabbix para:

- [brevis.one](#)
- [Discord](#)
- [Express.ms messenger](#)
- [Github issues](#)
- [iLert](#)
- [iTop](#)
- [Jira](#)
- [Jira Service Desk](#)
- [ManageEngine ServiceDesk](#)
- [Mattermost](#)
- [Microsoft Teams](#)
- [Opsgenie](#)
- [OTRS](#)
- [Pagerduty](#)
- [Pushover](#)
- [Redmine](#)
- [Rocket.Chat](#)
- [ServiceNow](#)
- [SIGNL4](#)
- [Slack](#)
- [SolarWinds](#)
- [SysAid](#)
- [Telegram](#)
- [TOPdesk](#)
- [VictorOps](#)
- [Zammad](#)
- [Zendesk](#)

Note:

Em adição aos serviços listados aqui, o Zabbix pode ser integrado com **Spiceworks** (nenhum webhook é necessário). Para converter notificações Zabbix em chamados do Spiceworks, crie um **tipo de mídia email** e informe o endereço de e-mail de helpdesk do Spiceworks (p.e. help@zabbix.on.spiceworks.com) nas configurações de perfil de um usuário Zabbix designado.

Configuração

Para começar a usar uma integração webhook:

1. Localize o arquivo .xml requerido no diretório `templates/media` da versão de Zabbix baixada ou baixe-o do [repositório git](#) do Zabbix
2. **Importe** o arquivo para dentro de sua instalação do Zabbix. O webhook aparecerá nas lista de tipos de mídia.
3. Configure o webhook de acordo com as instruções no arquivo `Readme.md` (você pode clicar no nome do webhook acima para acessar rapidamente o `Readme.md`).

Para criar um webhook customizado do zero:

- Vá até Administração → Tipos de mídia
- Clique em Criar tipo de mídia

A aba **Tipo de mídia** contém vários atributos específicos para este tipo de mídia:

* Name Pushover

Type Webhook

Parameters	Name	Value	Action
	endpoint	https://api.pushover.net/1/message	Remove
	eventid	{EVENT.ID}	Remove
	expire	1200	Remove
	message	{ALERT.MESSAGE}	Remove
	priority	0	Remove
	retry	60	Remove
	title	{ALERT.SUBJECT}	Remove
	token	<PUSHOVER TOKEN HERE>	Remove
	triggerid	{TRIGGER.ID}	Remove
	url	{ZABBIX.URL}	Remove
	url_title	Zabbix	Remove
	user	{ALERT.SENDTO}	Remove
	Add		

* Script try {...

Timeout 30s

Process tags ☐Include event menu entry ☐

* Menu entry name

* Menu entry URL

Description Please refer to setup guide here: <https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/media/pushover>

Set token parameter to your Pushover application key.
When assigning Pushover media to the Zabbix user - add the user key into send to field.

Enabled ☒

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Os seguintes parâmetros são específicos para o tipo de mídia webhook:

Parâmetro	Descrição
Parâmetros	<p>Especifique as variáveis de webhook como pares de atributo e valor.</p> <p>Para webhooks pré-configurados, a lista de parâmetros varia, dependendo do serviço. Verifique o arquivo Readme.md do webhook para descrição de parâmetro.</p> <p>Para novos webhooks, várias variáveis comuns estão incluídas por padrão (URL:<vazio>, HTTPProxy:<vazio>, To:{ALERT.SENDTO}, Subject:{ALERT.SUBJECT}, Message:%7BALERT.MESSAGE), sintaxe à vontade para manter ou removê-las.</p> <p>Todas as macros que são suportadas nas notificações de problema são suportadas nos parâmetros.</p> <p>Se você especificar um proxy HTTP, o campo suporta as mesmas funcionalidades que o campo HTTP proxy da configuração de item. A string de proxy pode ser prefixada com <code>[scheme]://</code> para especificar qual tipo de proxy é usado (p.e. https, socks4, socks5; veja documentação).</p>
Script	<p>Informe o código JavaScript no bloco que aparece quando clicando no campo do parâmetro (ou no botão ver/editar próximo a ele). Este código executará a operação de webhook.</p> <p>O script é um código de função que aceita parâmetro - pares de valor. Os valores devem ser convertidos em objetos JSON usando o método <code>JSON.parse()</code>, por exemplo: <code>var params = JSON.parse(value);</code>.</p> <p>O código tem acesso a todos os parâmetros, ele pode executar requisições HTTP de GET, POST, PUT e DELETE e tem controle sobre cabeçalhos e corpo de requisição HTTP.</p> <p>O script deve conter um operador de retorno, caso contrário ele não será válido. Ele pode retornar o estado OK em conjunto com uma lista opcional de etiquetas e valores de etiqueta (veja a opção Etiquetas de processo) ou uma string de erro.</p> <p>Note que o script é executado somente após um alerta ser criado. Se o script for configurado para retornar e processar etiquetas, estas etiquetas não serão resolvidas nas macros <code>{EVENT.TAGS}</code> e <code>{EVENT.RECOVERY.TAGS}</code> na mensagem de problema inicial e mensagens de recuperação porque o script ainda não teve tempo de ser executado.</p> <p>Veja também: Diretrizes de desenvolvimento de Webhook, Exemplos de script de webhook, Objetos JavaScript adicionais.</p>
Tempo limite	<p>Tempo limite de execução do JavaScript (1-60s, padrão é 30s). Sufixos de tempo são suportados, p.e. 30s, 1m.</p>
Etiquetas de processo*	<p>Marque a caixa para processar os valores de propriedade JSON retornados como etiquetas. Estas etiquetas são adicionadas às etiquetas de evento de problema já existentes (se houver) no Zabbix.</p> <p>Se um webhook usa etiquetas (a caixa Processar etiquetas estiver marcada), o webhook deve sempre retornar um objeto JSON contendo ao menos um objeto vazio para etiquetas: <code>var result = {tags: {}};</code>.</p> <p>Exemplos de etiquetas que podem ser retornadas: Jira ID: PROD-1234, Responsável: John Smith, Processado:<sem valor>, etc.</p>

Parâmetro	Descrição
Incluir entrada de menu de evento	Marque a caixa para incluir uma entrada no menu de evento associando-o ao chamado externo criado. Se marcado, o webhook não deve ser usado para enviar notificações para usuários diferentes (considere a criação de um usuário dedicado em vez disso) ou em várias ações de alerta relacionadas a um único evento de problema .
Nome da entrada de menu	Especifique o nome da entrada de menu. A macro {EVENT.TAGS.<nome etiqueta>} é suportada. Este campo é obrigatório apenas se Incluir entrada de menu de evento estiver selecionado.
URL de entrada de menu	Especifique a URL associada à entrada de menu. A macro {EVENT.TAGS.<nome etiqueta>} é suportada. Este campo é obrigatório apenas se Incluir entrada de menu de evento estiver selecionado.

Consulte os **parâmetros de tipo de mídia comuns** para detalhes em como configurar opções de mensagens padrão e processamento de alerta.

Warning:

Mesmo que um webhook não use mensagens padrão, modelos de mensagem para os tipos de operação usados por este webhook ainda devem ser definidos.

Mídia de usuário

Uma vez que o tipo de mídia é configurado, vá até a seção Administração → Usuários e associe a mídia webhook para um usuário existente ou crie um novo usuário para representar o webhook. As etapas para configuração de mídia de usuário para um usuário existente, sendo comum para todos os tipos de mídia, estão descritas na página **Tipos de mídia**.

Se um webhook usa etiquetas para armazenar o ID de chamado\mensagem, evite associar o mesmo webhook como um mídia para diferentes usuários pois fazendo isto pode causar erros no webhook (se aplica à maioria dos webhooks que utilizam a opção Incluir entrada de menu de evento). Neste caso, a melhor prática é criar um usuário dedicado para representar o webhook:

1. Após a configuração do tipo de mídia webhook, vá até a seção Administração → Usuários e crie um usuário Zabbix dedicado para representar o webhook - por exemplo, com um usuário Slack para o webhook Slack. Todas as configurações, exceto mídia, podem ser deixadas em seu padrão pois este usuário não fará login no Zabbix.
2. No perfil de usuário, vá até a aba Mídia e **adicione um webhook** com as informações de contato necessárias. Se o webhook não usa um campo Enviar para, informe qualquer combinação de caracteres suportados para atender à validação de requisitos.
3. Garante a este usuário ao menos **permissões** de leitura para todos os hosts para os quais ele deve enviar os alertas.

Quando configurando um ação de alerta, adicione este usuário no campo Enviar para usuários nos detalhes de Operação - isto informará ao Zabbix para usar o webhook para notificação desta ação.

Configurando ações de alerta

Ações determinam que notificações devem ser enviadas através do webhook. As etapas para **configuração de ações** envolvendo webhooks são as mesmas para todos os tipos de mídia com estas exceções:

- Se um webhook usa etiquetas para armazenar ID de chamado\mensagem e para acompanhar operações de atualização\solução, este webhook não deve ser usado em várias ações de alerta para um único evento de problema. Se {EVENT.TAGS.<nome>} já existir, e estiver atualizado no webhook, então seu valor resultante não é definido. Para tal caso, um novo nome de etiqueta deve ser usado no webhook para armazenar valores atualizados. Isto se aplica aos webhooks do Jira, Jira Service Desk, Mattermost, Opsgenie, OTRS, Redmine, ServiceNow, Slack, Zammad, e Zendesk fornecidos pela Zabbix e à maioria dos webhooks que utilizam a opção Incluir entrada no menu de evento. A utilização do webhook em várias operações é permitida se essas operações ou etapas de escalção pertencerem à mesma ação. É também possível usar este webhook em diferentes ações se as ações não serão aplicadas ao mesmo evento de problema devido condições de filtro diferentes.
- Quando usando um webhook em ações para **eventos internos**: na configuração da operação da ação, verifique a caixa Mensagem customizada e defina a mensagem customizada, caso contrário, não será enviada uma notificação.

Exemplos de script de webhook

Visão geral

Apesar de o Zabbix oferecer um grande número de integrações webhook disponíveis de forma nativa, você pode querer criar seus próprios webhooks. Esta seção provê exemplos de scripts webhook customizados (usados no parâmetro Script). Consulte a seção [webhook](#) para descrição de outros parâmetros de webhook.

Webhook para Jira (customizado)

Media type
Message templates 5
Options

* Name
Jira webhook

Type
Webhook

Parameters

Name	Value
HTTPProxy	
Message	{ALERT.MESSAGE}
Subject	{ALERT.SUBJECT}
To	{ALERT.SENDTO}
URL	

Add

* Script
try {...

* Timeout
30s

Process tags
☒

Include event menu entry
☒

* Menu entry name
{EVENT.tags.issue_key}

* Menu entry URL
https://tsupport.zabbix.lan/browse/{EVENT.tags.issue_key}

Description
Creating a JIRA issue.

Enabled
☒

Este script criará um chamado (issue) no JIRA e retornará informações sobre o chamado criado.

```
try {
  Zabbix.log(4, '[ Jira webhook ] Iniciado com params: ' + value);

  var result = {
    'tags': {
      'endpoint': 'jira'
    }
  },
  params = JSON.parse(value),
  req = new HttpRequest(),
  fields = {},
  resp;
```

```

if (params.HTTPProxy) {
    req.setProxy(params.HTTPProxy);
}

req.addHeader('Content-Type: application/json');
req.addHeader('Authorization: Basic ' + params.authentication);

fields.summary = params.summary;
fields.description = params.description;
fields.project = {key: params.project_key};
fields.issuetype = {id: params.issue_id};

resp = req.post('https://tsupport.zabbix.lan/rest/api/2/issue/',
    JSON.stringify({"fields": fields})
);

if (req.getStatus() != 201) {
    throw 'Codigo de resposta: ' + req.getStatus();
}

resp = JSON.parse(resp);
result.tags.issue_id = resp.id;
result.tags.issue_key = resp.key;

return JSON.stringify(result);
}
catch (error) {
    Zabbix.log(4, '[ Jira webhook ] Criacao de chamado falhou (json) : ' + JSON.stringify({"fields": fields}));
    Zabbix.log(3, '[ Jira webhook ] Criacao de chamado falhou : ' + error);

    throw 'Falhou com erro: ' + error;
}

```

Webhook para Slack (customizado)

Este webhook encaminhará notificações do Zabbix para um canal do Slack.

```

try {
    var params = JSON.parse(value),
        req = new HttpRequest(),
        response;

    if (params.HTTPProxy) {
        req.setProxy(params.HTTPProxy);
    }

    req.addHeader('Content-Type: application/x-www-form-urlencoded');

    Zabbix.log(4, '[ Slack webhook ] Requisicao webhook com valor=' + value);

    response = req.post(params.hook_url, 'payload=' + encodeURIComponent(value));
    Zabbix.log(4, '[ Slack webhook ] Respondido com codigo: ' + req.Status() + '. Resposta: ' + response);

    try {
        response = JSON.parse(response);
    }
    catch (error) {
        if (req.getStatus() < 200 || req.getStatus() >= 300) {
            throw 'Requisicao falhou com codigo de estado ' + req.getStatus();
        }
        else {
            throw 'Requisicao bem-sucedida, mas analise da resposta falhou.';
        }
    }
}

```

```

    }

    if (req.getStatus() !== 200 || !response.ok || response.ok === 'false') {
        throw response.error;
    }

    return 'OK';
}
catch (error) {
    Zabbix.log(3, '[ Jira webhook ] Envio falhou. Erro: ' + error);

    throw 'Falhou com erro: ' + error;
}

```

2 Ações

Visão geral

Se você quiser que algumas operações ocorram como resposta a eventos (por exemplo, notificações enviadas), você precisa configurar ações.

Ações podem ser definidas em resposta a eventos de todos os tipos suportados:

- Ações de gatilho - para eventos quando o estado de um gatilho muda de OK para PROBLEMA e vice-versa
- Ações de serviço - para eventos quando o estado de um serviço muda de OK para PROBLEMA e vice-versa
- Ações de descoberta - para eventos quando uma descoberta de rede ocorre
- Ações de autoregistro - para eventos quando um novo agente ativo se autoregistra (ou os metadados de um host registrado sofre alteração)
- Ações internas - para eventos quando itens se tornam não suportados ou gatilhos entram em um estado desconhecido

Configurando uma ação

Para configurar uma ação, faça o seguinte:

- Vá até Configuração -> Ações e selecione o tipo de ação requerido a partir do submenu (você pode alterar para outro tipo mais tarde, usando a lista de títulos)
- Clique em Criar ação
- Nomeie a ação
- Escolha as **condições** nas quais as operações são empreendidas
- Escolha as **operações** a serem empreendidas

Note que ações de serviço podem ser configuradas na seção **ação de serviço**.

Atributos de ação gerais:

Action

Operations

* Name

Report problems to Zabbix administrators

Type of calculation

And

A and B

Conditions

Label	Name
A	Trigger severity is greater than or equals <i>Not classified</i>
B	Trigger severity does not equal <i>Information</i>
<div>Add</div>	

Enabled

☒

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	Nome de ação único.
Tipo de cálculo	Selecione a opção de avaliação para as condições da ação (com mais de uma condição): And - todas as condições devem ser atendidas Or - basta uma condição ser atendida And/Or - combinação das duas: AND com diferentes tipos de condição e OR com o mesmo tipo de condição Custom expression - uma fórmula de cálculo definida pelo usuário para avaliação das condições da ação.
Condições	Lista de condições da ação. Clique em Adicionar para adicionar uma nova condição .
Habilitado	Marque a caixa de seleção para habilitar a ação. Caso contrário, ela estará desabilitada.

1 Condições

Visão geral

É possível definir que uma ação seja executada apenas se o evento corresponde a um conjunto de condições definidas. Condições são definidas quando configurando a **ação**.

A correspondência de condições é sensível a letras maiúsculas e minúsculas.

Ações de gatilho

As seguintes condições podem ser usadas em ações baseadas em gatilho:

Tipo de condição	Operadores suportados	Descrição
Grupo de host	igual (equals) não igual (does not equal)	Especifica grupos de host ou grupos de host para exclusão. equals - evento pertence a este grupo de host. does not equal - evento não pertence a este grupo de host. A especificação de um grupo de host pai seleciona implicitamente todos os grupos aninhados. Para especificar apenas o grupo pai, todos os grupos aninhados têm de ser adicionalmente configurados com o operador does not equal .
Modelo (template)	igual não igual	Especifica modelos ou modelos para exclusão. equals - evento pertence a um gatilho herdado deste modelo. does not equal - evento não pertence a um gatilho herdado deste modelo.
Host	igual não igual	Especifica hosts ou hosts para exclusão. equals - evento pertence a este host. does not equal - evento não pertence a este host.

Tipo de condição	Operadores suportados	Descrição
Nome da etiqueta	igual não igual contém (contains) não contém (does not contain)	Especifica etiqueta de evento ou etiqueta de evento para exclusão. equals - evento possui esta etiqueta does not equal - evento não possui esta etiqueta contains - evento possui uma etiqueta contendo esta string does not contain - evento não possui uma etiqueta contendo esta string
Valor da etiqueta	igual não igual contém não contém	Especifica etiqueta de evento e combinação de valor ou etiqueta e combinação de valor para exclusão. equals - evento possui esta etiqueta e valor does not equal - evento não possui esta etiqueta e valor contains - evento possui uma etiqueta e valor contendo estas strings does not contain - evento não possui uma etiqueta e valor contendo estas strings
Gatilho	igual não igual	Especifica gatilhos e gatilhos para exclusão. equals - evento é gerado por este gatilho. does not equal - evento é gerado por qualquer outro gatilho, exceto este.
Nome do gatilho	contém não contém	Especifica uma string no nome do gatilho ou uma string para exclusão. contains - evento é gerado por um gatilho, contendo esta string no nome. does not contain - esta string não pode ser encontrada no nome do gatilho. Nota: O valor informado será comparado ao nome do gatilho com todas as macros expandidas.
Severidade do gatilho	igual não igual é maior que (is greater than) ou igual (equals) é menor que (is less than) ou igual (equals)	Especifica severidade do gatilho. equals - igual à severidade do gatilho does not equal - não igual à severidade do gatilho is greater than or equals - maior ou igual à severidade do gatilho is less than or equals - menor ou igual à severidade do gatilho

Tipo de condição	Operadores suportados	Descrição
Período de tempo	dentro (in) fora (not in)	<p>Especifica um período de tempo ou um período de tempo para exclusão.</p> <p>in - horário do evento está dentro do período de tempo.</p> <p>not in - horário do evento está fora do período de tempo.</p> <p>Consulte a página de especificação de período de tempo para descrição do formato.</p> <p>Macros de usuário são suportadas, desde o Zabbix 3.4.0.</p>
Problema é suprimido	não (no) sim (yes)	<p>Especifica se o problema é suprimido (não mostrado) devido manutenção do host.</p> <p>no - problema não é suprimido.</p> <p>yes - problema é suprimido.</p>

Ações de descoberta

As seguintes condições podem ser usadas em eventos baseados em descoberta:

Tipo de condição	Operadores suportados	Descrição
IP do host	igual (equals) não igual (does not equal)	<p>Especifica um intervalo de endereço IP ou um intervalo para exclusão para um host descoberto.</p> <p>equals - IP do host está no intervalo.</p> <p>does not equal - IP do host não está no intervalo.</p> <p>Ele pode ter os seguintes formatos:</p> <p>IP único: 192.168.1.33</p> <p>Intervalo de endereços IP: 192.168.1-10.1-254</p> <p>Máscara IP: 192.168.4.0/24</p> <p>Lista: 192.168.1.1-254, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24</p> <p>Suporte para espaços no formato de lista é fornecido desde o Zabbix 3.0.0.</p>

Tipo de condição	Operadores suportados	Descrição
Tipo de serviço	igual não igual	<p>Especifica um tipo de serviço de um serviço descoberto ou um tipo de serviço para exclusão.</p> <p>equals - corresponde ao serviço descoberto.</p> <p>does not equal - não corresponde ao serviço descoberto.</p> <p>Tipos de serviço disponíveis: SSH, LDAP, SMTP, FTP, HTTP, HTTPS (disponível desde a versão 2.2 do Zabbix), POP, NNTP, IMAP, TCP, Zabbix agent, SNMPv1 agent, SNMPv2 agent, SNMPv3 agent, ICMP ping, telnet (disponível desde a versão 2.2 do Zabbix).</p>
Porta do serviço	igual não igual	<p>Especifica um intervalo de porta TCP de um serviço descoberto ou um intervalo para exclusão.</p> <p>equals - porta do serviço está no intervalo.</p> <p>does not equal - porta do serviço não está no intervalo.</p>
Regra de descoberta	igual não igual	<p>Especifica uma regra de descoberta ou uma regra de descoberta para exclusão.</p> <p>equals - usando esta regra de descoberta.</p> <p>does not equal - usando qualquer outra regra de descoberta, exceto esta.</p>
Verificação de descoberta	igual não igual	<p>Especifica uma verificação de descoberta ou uma verificação de descoberta para exclusão.</p> <p>equals - usando esta verificação de descoberta.</p> <p>does not equal - usando qualquer outra verificação de descoberta, exceto esta.</p>
Objeto da descoberta	igual (equals)	<p>Especifica o objeto da descoberta.</p> <p>equals - igual ao objeto descoberto (um dispositivo ou um serviço).</p>
Estado da descoberta	igual	<p>Up - corresponde aos eventos 'Host Up' e 'Serviço Up'</p> <p>Down - corresponde aos eventos 'Host Down' e 'Serviço Down'</p> <p>Discovered - corresponde aos eventos 'Host Descoberto' e 'Serviço Descoberto'</p> <p>Lost - corresponde aos eventos 'Host Perdido' e 'Serviço Perdido'</p>

Tipo de condição	Operadores suportados	Descrição
Tempo ativo (Uptime)/Tempo inativo (Downtime)	é maior que ou igual é menor que ou igual	Tempo ativo para eventos 'Host Up' e 'Serviço Up'. Tempo inativo para eventos 'Host Down' e 'Serviço Down'. is greater than or equals - é maior ou igual a. Parâmetro é dado em segundos. is less than or equals - é menor ou igual a. Parâmetro é dado em segundos.
Valor recebido	igual não igual é maior que ou igual é menor que ou igual contém não contém	Especifica o valor recebido de uma verificação de agente (Zabbix, SNMP) em uma regra de descoberta. Comparação de string. Se várias verificações de Zabbix Agent ou SNMP são definidas para uma regra, os valores recebidos para cada uma delas são verificadas (cada verificação gera um novo evento que é correspondido contra todas as condições). equals - igual ao valor. does not equal - não igual ao valor. is greater than or equals - maior ou igual ao valor. is less than or equals - menor ou igual ao valor. contains - contém a substring. Parâmetro é dado como uma string. does not contain - não contém a substring. Parâmetro é dado como uma string.
Proxy	igual não igual	Especifica um proxy ou um proxy para exclusão. equals - usando este proxy. does not equal - não usando qualquer outro proxy exceto este.

Note:

Verificações de serviço em uma regra de descoberta, que resultam em eventos de descoberta, não ocorrem simultaneamente. Portanto, se **múltiplos** valores são configurados para condições de Tipo do serviço, Porta do serviço ou Valor recebido na ação, eles serão comparados com um evento de descoberta por vez, mas **não** com vários eventos simultaneamente. Como resultado, ações com múltiplos valores para os mesmos tipos de verificação podem não ser executadas corretamente.

Ações de autoregistro

As seguintes condições podem ser usadas em ações baseadas em autoregistro de agente ativo:

Tipo de condição	Operadores suportados	Descrição
Metadados do host	contém não contém corresponde não corresponde	Especifica metadados do host ou metadados do host para exclusão. contains - metadados do host contém a string. does not contain - metadados do host não contém a string. Os metadados do host podem ser especificados em um arquivo de configuração . matches - metadados do host corresponde à expressão regular. does not match - metadados do host não corresponde à expressão regular.
Nome do host	contém não contém corresponde não corresponde	Especifica um nome de host ou um nome de host para exclusão. contains - nome do host contém a string. does not contain - nome do host não contém a string. matches - nome do host corresponde à expressão regular. does not match - nome do host não corresponde à expressão regular.
Proxy	igual não igual	Especifica um proxy ou um proxy para exclusão. equals - usando o proxy. does not equal - usando qualquer outro proxy exceto este.

Ações de evento interno

As seguintes condições podem ser definidas baseadas em eventos internos:

Tipo de condição	Operadores suportados	Descrição
Tipo do evento	igual (equals)	Item in "not supported" state - corresponde a eventos onde um item vai do estado 'normal' ao 'não suportado' Low-level discovery rule in "not supported" state - corresponde a eventos onde uma regra de descoberta de baixo-nível vai do estado 'normal' ao 'não suportado' Trigger in "unknown" state - corresponde a eventos onde um gatilho vai do estado 'normal' ao 'não suportado'

Tipo de condição	Operadores suportados	Descrição
Grupo de host	igual não igual	Especifica grupos de host ou grupos de host para exclusão. equals - evento pertence a este grupo de host. does not equal - evento não pertence a este grupo de host.
Nome de etiqueta	igual não igual contém não contém	Especifica etiqueta de evento ou etiqueta de evento para exclusão. equals - evento possui esta etiqueta does not equal - evento não possui esta etiqueta contains - evento possui uma etiqueta contendo esta string does not contain - evento não possui uma etiqueta contendo esta string
Valor da etiqueta	igual não igual contém não contém	Especifica combinação de etiqueta do evento e valor ou combinação de etiqueta e valor para exclusão. equals - evento possui esta etiqueta e valor does not equal - evento não possui esta etiqueta e valor contains - evento possui uma etiqueta e valor contendo estas strings does not contain - evento não possui uma etiqueta e valor contendo estas strings
Modelo	igual não igual	Especifique modelos ou modelos para exclusão. equals - evento pertence a um item/gatilho/regra de descoberta de baixo-nível herdado deste modelo. does not equal - evento não pertence a um item/gatilho/regra de descoberta de baixo-nível herdado deste modelo.
Host	igual não igual	Especifica hosts ou hosts para exclusão. equals - evento pertence a este host. does not equal - evento não pertence a este host.

Tipo de cálculo

As seguintes opções de cálculo de condições estão disponíveis:

- **And** - todas as condições devem ser atendidas

Note que o uso de cálculo "And" não é permitido entre vários gatilhos quando eles são selecionados como uma condição Trigger=. Ações só podem ser executadas baseadas no evento de um gatilho.

- **Or** - suficiente se uma condição é atendida
- **And/Or** - combinação das duas: AND com diferentes tipos de condição e OR com o mesmo tipo de condição, por exemplo:

Grupo de host igual Oracle servers
Grupo de host igual MySQL servers
Nome do gatilho contém 'Banco de dados está down'
Nome do gatilho contém 'Banco de dados está indisponível'

é avaliado como

(Grupo de host equals Oracle servers **or** Grupo de host equals MySQL servers) **and** (Nome do gatilho contains 'Banco de dados está down' **or** Nome do gatilho contains 'Banco de dados está indisponível')

- **Custom expression** - (expressão customizada) uma fórmula de cálculo definida pelo usuário para avaliação de condições de ação. Deve incluir todas as condições (representadas como letra maiúsculas A, B, C, ...) e pode incluir espaços, tabulações, parêntesis (), **and** (sensível a maiúsculas e minúsculas), **or** (sensível a maiúsculas e minúsculas), **not** (sensível a maiúsculas e minúsculas).

Enquanto o exemplo anterior com **And/Or** seria representado como (A or B) e (C or D), em uma expressão customizada você pode também ter múltiplas formas de cálculo:

(A and B) and (C or D)
(A and B) or (C and D)
((A or B) and C) or D
(not (A or B) and C) or not D
etc.

Ações desabilitadas devido objetos eliminados

Se um certo objeto (host, modelo, gatilho, etc.) usado em uma ação condição/operação é eliminado, a condição/operação é removida e a ação é desabilitada para evitar execução incorreta da ação. A ação pode ser reabilitada pelo usuário.

Este comportamento ocorre quando eliminando:

- grupos de host (condição "grupo de host", operação "comando remoto" em um grupo de host específico);
- hosts (condição "host", operação "comando remoto" em um host específico);
- modelos (condição "modelo", operações "associar a modelo" e "desassociar de modelo");
- gatilhos (condição "gatilho");
- regras de descoberta (quando usando condições "regra de descoberta" e "verificação de descoberta").

Nota: Se um comando remoto tem muitos hosts alvo, e nós eliminamos um deles, apenas este host será removido da lista alvo, a operação em si permanecerá. Mas, se for o único host, a operação será removida, também. O mesmo se aplica para operações "associar a modelo" e "desassociar de modelo".

Ações não são desabilitadas quando eliminando um usuário ou grupo de usuário usado em uma operação "enviar mensagem".

2 Operações

Visão geral

Você pode definir as seguintes operações para todos os eventos:

- enviar uma mensagem
- executar um comando remoto

Attention:

O servidor Zabbix Server não cria alertas se o acesso ao host é explicitamente "negado" para o usuário definido como destinatário da operação da ação ou se o usuário não tem direitos definidos para o host.

Para eventos de descoberta e autoregistro, há operações adicionais disponíveis:

- adicionar host
- remover host
- habilitar host
- desabilitar host
- adicionar ao grupo de host
- remover do grupo de host
- associar ao template
- desassociar do template
- definir modo de inventário do host

Configurando uma operação

Para configurar uma operação, vá para a aba Operações na configuração da ação.

Action

Operations 2

* Default operation step duration

1h

Operations

Steps Details

1 Send message to user groups: Zabbix administrators vi

Add

Recovery operations

Details

Notify all involved

Add

Action

Edit

Update operations

Details

Add

Action

Pause operations for suppressed problems

☒

Notify about canceled escalations

☒

* At least one operation must exist.

Add

Cancel

Atributos de operação geral:

Parâmetro	Descrição
Duração da etapa de operação padrão	Duração de uma etapa de operação por padrão (60 segundos à 1 semana). Por exemplo, uma duração de etapa de uma hora significa que se uma operação é realizada, uma hora passará antes da nova etapa. Os Sufixos de tempo são suportados, p.e. 60s, 1m, 2h, 1d, desde o Zabbix 3.4.0. Macros de usuário são suportadas, desde o Zabbix 3.4.0.
Operações	Operações de ação (se houver) são exibidas, com estes detalhes: Steps - etapa(s) de escalação à qual a operação é associada Details - tipo de operação e seus destinatários/alvo. A lista de operação também exibe o tipo de mídia (e-mail, SMS ou script) usado assim como o nome e sobrenome (em parênteses após o nome de usuário) de um destinatário de notificação. Start in - quanto tempo após um evento a operação é executada Duration (sec) - duração da etapa é exibida. Default (padrão) é exibido se uma etapa usa a duração padrão, e um horário é exibido se uma duração customizada é utilizada. Action - links para edição e remoção de uma operação são exibidos.

Parâmetro	Descrição
Operações de recuperação	Operações de ação (se houver) são exibidas, com estes detalhes: Details - tipo de operação e seus destinatários/alvo. A lista de operação também exibe o tipo de mídia (e-mail, SMS ou script) usado assim como o nome e sobrenome (em parênteses após o nome de usuário) de um destinatário de notificação. Action - links para edição e remoção de uma operação são exibidos.
Operações de atualização	Operações de ação (se houver) são exibidas, com estes detalhes: Details - tipo de operação e seus destinatários/alvo. A lista de operação também exibe o tipo de mídia (e-mail, SMS ou script) usado assim como o nome e sobrenome (em parênteses após o nome de usuário) de um destinatário de notificação. Action - links para edição e remoção de uma operação são exibidos.
Pausa operações para problemas suprimidos	Marque esta caixa de seleção para atrasar o início das operações pela duração de um período de manutenção. Quando as operações são iniciadas, após a manutenção, todas as operações são executadas incluindo aquelas para os eventos durante a manutenção. Note que esta configuração afeta apenas escalações de problema; operações de recuperação e atualização não serão afetadas. Se você desmarcar esta caixa, as operações serão executadas sem atraso mesmo durante um período de manutenção. Esta opção não está disponível para Ações de serviço.
Notificar sobre escalações canceladas	Desmarque esta caixa de seleção para desabilitar notificações sobre escalações canceladas (quando um host, item, gatilho ou ação é desabilitado).

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho

Para configurar os detalhes de uma nova operação, clique em [Add](#) no bloco Operações. Para editar uma operação existente, clique em [Edit](#) próximo à operação. Uma janela abrirá onde você pode editar os detalhes da etapa de operação.

Detalhes de operação

Operation details

×

Operation

Send message

▼

Steps

1

-

1

(0 - infinitely)

Step duration

0

(0 - use action default)

* At least one user or user group must be selected.

Send to user groups

User group	Action
Zabbix administrators	Remove
Add	

Send to users

User	Action
Add	

Send only to

Email

▼

Custom message

☐

Conditions

Label	Name	Action
A	Event is not acknowledged	Remove
Add		

Update

Cancel

Operation	Send message	▼
-----------	--------------	---

Steps	1	-	1	(0 - infinitely)
-------	---	---	---	------------------

Step duration	0	(0 - use action default)
---------------	---	--------------------------

* At least one user or user group must be selected.

Send to user groups

User group

Zabbix administrators

Add

Action

Remove

Send to users

User

Add

Action

Send only to	Email
--------------	-------

Email

Custom message ☐

Conditions

Label

Name

A

Event is not acknowledged

Add

Action

Remove

Update

Cancel

Parâmetro	Descrição
Operação	<p>Selecione a operação:</p> <p>Send message - enviar mensagem para o usuário <nome comando remoto> - executa um comando remoto. Comandos estão disponíveis para execução se previamente definidos em scripts globais com Operação de ação selecionado como seu escopo. Mais operações estão disponíveis para eventos baseados em descoberta e autoregistro (veja acima).</p>
Etapas	<p>Selecione a(s) etapa(s) à(s) qual(is) associar um planejamento de escalação:</p> <p>From - execute iniciando com esta etapa To - execute até esta etapa (0=infinito, execução não será limitada)</p>

Parâmetro	Descrição
Duração da etapa	<p>Duração customizada para estas etapas (0=usa duração de etapa padrão).</p> <p>Sufixos de tempo são suportados, e.g. 60s, 1m, 2h, 1d, desde o Zabbix 3.4.0.</p> <p>Macros de usuário são suportadas, desde o Zabbix 3.4.0.</p> <p>Várias operações podem ser associadas à mesma etapa. Se estas operações têm diferentes durações de etapa definidas, a mais curta é considerada e aplicada à etapa.</p>
Tipo de operação: enviar mensagem	
Enviar para grupos de usuário	<p>Clique em Adicionar para selecionar grupos de usuário aos quais enviar mensagens.</p> <p>O grupo de usuário deve ter ao menos permissões de "leitura" ao host de modo a ser notificado.</p>
Enviar para usuários	<p>Clique em Adicionar para selecionar usuários aos quais enviar mensagens.</p> <p>O usuário deve ter ao menos permissões de "leitura" ao host de modo a ser notificado.</p>
Enviar apenas para Mensagem customizada	<p>Enviar mensagem para todos os tipos de mídia definidos ou apenas um selecionado.</p>
	<p>Se selecionado, a mensagem customizada pode ser configurada.</p> <p>Para notificações sobre eventos internos via webhooks, uma mensagem customizada é obrigatória.</p>
Assunto	Assunto da mensagem customizada. O assunto pode conter macros. É limitado a 255 caracteres.
Mensagem	A mensagem customizada. A mensagem pode conter macros. É limitada a uma certa quantidade de caracteres dependendo do tipo de banco de dados (veja Envio de mensagem para mais informações).
Tipo de operação: comando remoto	

Parâmetro	Descrição
Lista alvo	<p>Selecione alvos nos quais executar o comando:</p> <p>Current host - comando é executado no host do gatilho que causou o evento de problema. Esta opção não funcionará se houver múltiplos hosts no gatilho.</p> <p>Host - selecione o(s) host(s) no(s) qual(is) executar o comando.</p> <p>Grupo de host - selecione o(s) grupo(s) de host(s) no(s) qual(is) executar o comando. A especificação de um grupo de host pai seleciona implicitamente todos os grupos aninhados. Assim o comando remoto será executado nos hosts dos grupos aninhados.</p> <p>Um comando em um host é executado apenas uma vez, mesmo que haja múltipla correspondência para o host (p.e. de vários grupos de host; individualmente e de um grupo de host).</p> <p>A lista alvo é insignificante se um script customizado é executado no Zabbix Server. A seleção de mais alvos neste caso resulta apenas no script sendo executado mais vezes no Server.</p> <p>Note que para scripts globais, a seleção do alvo também depende da definição do Grupo de host na configuração de script global.</p> <p>A opção Lista alvo não está disponível para Ações de serviço porque neste caso os comandos remotos são sempre executados no Zabbix Server.</p> <p>Condição para execução da operação:</p> <p>Not ack - apenas quando o evento é não reconhecido (unacknowledged)</p> <p>Ack - apenas quando o evento é reconhecido (acknowledged).</p> <p>A opção Condições não está disponível para Ações de serviço.</p>
Condições	

Quando estiver pronto, clique em Adicionar para adicionar a operação à lista de Operações.

1 Enviando mensagem

Visão geral

Enviar uma mensagem é uma das melhores maneiras de notificar as pessoas sobre um problema. Por isso é uma das principais ações oferecidas pelo Zabbix.

Configuração

Para poder enviar e receber notificações do Zabbix você precisa:

- **definir a mídia** para enviar um mensagem para

A gravidade do gatilho padrão ('Não classificado') **deve ser** verificado na mídia do usuário **configuração** se você deseja receber notificações para eventos não acionadores, como descoberta, registro automático de agente ativo ou eventos internos.

- **configurar uma ação operação** que envia uma mensagem para uma das mídias definidas

::: não importante Zabbix envia notificações apenas para esses usuários que tenham pelo menos permissões de 'leitura' para o host que gerou o evento. Pelo menos um host de uma expressão de gatilho deve ser acessível. :::

Você pode configurar cenários personalizados para enviar mensagens usando **escalonamento**.

Para receber e ler e-mails do Zabbix com sucesso, servidores/clientes devem suportar o formato padrão 'SMTP/MIME e-mail', pois Zabbix envia dados UTF-8 (Se o assunto contiver apenas caracteres ASCII, não é codificado em UTF-8.). O assunto e o corpo da mensagem são codificado em base64 para seguir o padrão de formato 'SMTP/MIME e-mail'.

O limite de mensagens após a expansão de todas as macros é igual ao limite de mensagens para **Remoto comandos**.

Mensagens de rastreamento

Você pode visualizar o status das mensagens enviadas em Monitoramento → Problemas.

Na coluna Ações você pode ver informações resumidas sobre as ações levado. Lá os números verdes representam as mensagens enviadas, os vermelhos - falha mensagens. Em andamento indica que uma ação foi iniciada. Fracassado informa que nenhuma ação foi executada com sucesso.

Se você clicar no horário do evento para ver os detalhes do evento, você também verá o bloco Ações de mensagens contendo detalhes das mensagens enviadas (ou não enviado) devido ao evento.

Em Relatórios → Registro de ações você verá detalhes de todas as ações realizadas para aqueles eventos que possuem uma ação configurada.

2 Comandos remotos

Visão geral

Com comandos remotos você pode definir que um determinado comando pré-definido é executado automaticamente no host monitorado sob alguma condição.

Assim, os comandos remotos são um mecanismo poderoso para inteligência proativa monitoramento.

Nos usos mais óbvios do recurso, você pode tentar:

- Reinicie automaticamente algum aplicativo (servidor web, middleware, CRM) se não responder
- Use o comando IPMI 'reboot' para reinicializar algum servidor remoto se isso acontecer não atende pedidos
- Liberar automaticamente o espaço em disco (removendo arquivos antigos, limpando /tmp) se ficar sem espaço em disco
- Migrar uma VM de uma caixa física para outra dependendo da CPU carga
- Adicione novos nós a um ambiente de nuvem em caso de CPU insuficiente (disco, memória, o que for) recursos

A configuração de uma ação para comandos remotos é semelhante à de envio uma mensagem, a única diferença é que o Zabbix irá executar um comando em vez de enviar uma mensagem.

Comandos remotos podem ser executados pelo servidor Zabbix, proxy ou agente. Controle remoto comandos no agente Zabbix podem ser executados diretamente pelo servidor Zabbix ou através do proxy Zabbix. Tanto no Zabbix Agent quanto no Zabbix proxy remote comandos são desabilitados por padrão. Eles podem ser habilitados por:

- adicionar um parâmetro `AllowKey=system.run[*]` na configuração do agente;
- definindo o parâmetro `EnableRemoteCommands` para '1' no proxy configuração.

Os comandos remotos executados pelo servidor Zabbix são executados conforme descrito em **Execução do comando** incluindo saída verificação de código.

Os comandos remotos são executados mesmo se o host de destino estiver em manutenção.

Limite de comando remoto

O limite de comando remoto após a resolução de todas as macros depende do tipo de banco de dados e conjunto de caracteres (caracteres não ASCII requerem mais de um byte a ser armazenado):

Banco de dados	Limite em caracteres	Limite em bytes
MySQL	65535	65535
Banco de dados Oracle	2048	4000
PostgreSQL	65535	não limitado
SQLite (somente proxy Zabbix)	65535	não limitado

O tutorial a seguir fornece instruções passo a passo sobre como definir comandos remotos.

Configuração

Esses comandos remotos que são executados no agente Zabbix (scripts personalizados) deve ser ativado primeiro no agente **configuração**.

Certifique-se de que o parâmetro `AllowKey=system.run[*]` seja adicionado. Reiniciar daemon do agente se alterar este parâmetro.

::: não importante Comandos remotos não funcionam com Zabbix ativo agentes. :::

Então, ao configurar uma nova ação em Configuração → Ações:

- Definir as condições adequadas. Neste exemplo, defina que o ação é ativada em qualquer problema de desastre com um dos Apache formulários:

Conditions	
Label	Name
A	Problem is not suppressed
B	Application contains <i>Apache</i>
C	Trigger severity is greater than or equals <i>Disaster</i>
Add	

Enabled ☒

- No **Operações** guia, clique em Adicionar nas operações/operações de recuperação/atualização bloco de operações
- No campo suspenso Operação, selecione um dos pré-definidos roteiros

Operation: Restart webserver

Steps: Restart webserver

- Selecione a lista de destino para o script

Scripts predefinidos

Todos os scripts (webhook, script, SSH, Telnet, IPMI) disponíveis para operações de ação são definidas em **global scripts**.

Por exemplo:

```
sudo /etc/init.d/apache restart
```

Neste caso, o Zabbix tentará reiniciar um processo do Apache. Com isso comando, certifique-se de que o comando seja executado no agente Zabbix (clique o botão Agente Zabbix em Executar em).

::: não importante Observe o uso de **sudo** - o usuário do Zabbix não têm permissões para reiniciar os serviços do sistema por padrão. Veja abaixo para dicas sobre como configurar o **sudo**. :::

Note:

O agente Zabbix deve ser executado no host remoto e aceitar conexões de entrada. O agente Zabbix executa comandos em fundo.

Comandos remotos no agente Zabbix são executados sem timeout pelo `system.run[,nowait]` e não são verificados quanto aos resultados da execução. Em Servidor Zabbix e proxy Zabbix, comandos remotos são executados com timeout conforme definido no parâmetro `TrapperTimeout` de `zabbix_server.conf` ou `zabbix_proxy.conf` e são **marcado** para resultados de execução.

Permissões de acesso

Certifique-se de que o usuário 'zabbix' tenha permissões de execução para comandos. Alguém pode estar interessado em usar **sudo** para dar acesso a comandos privilegiados. Para configurar o acesso, execute como root:

```
#visudo
```

Linhas de exemplo que podem ser usadas no arquivo sudoers:

```
# permite que o usuário 'zabbix' execute todos os comandos sem senha.  
zabbix ALL=NOPASSWD: ALL
```

```
# permite que o usuário 'zabbix' reinicie o apache sem senha.  
zabbix ALL=NOPASSWD: /etc/init.d/apache restart
```

Em alguns sistemas, o arquivo sudoers impedirá que arquivos não locais usuários de executar comandos. Para mudar isso, comente **opção obrigatória** em /etc/sudoers.

Comandos remotos com múltiplas interfaces

Se o sistema de destino tiver várias interfaces do tipo selecionado (agente Zabbix ou IPMI), os comandos remotos serão executados no padrão interface.

É possível executar comandos remotos via SSH e Telnet usando outra interface que não a do agente Zabbix. A interface disponível para o uso é selecionado na seguinte ordem:

- * Interface padrão do agente Zabbix
- * Interface padrão SNMP
- * Interface padrão JMX
- * Interface padrão IPMI

Comandos remotos IPMI

Para comandos remotos IPMI, a seguinte sintaxe deve ser usada:

```
<comando> [<valor>]
```

Onde

- <command> - um dos comandos IPMI sem espaços
- <valor> - 'on', 'off' ou qualquer inteiro sem sinal. <valor> é um parâmetro opcional.

Exemplos

Exemplos de **global scripts** que podem ser usados como comandos remotos em operações de ação.

Exemplo 1

Reinicialização do Windows em determinadas condições.

Para reiniciar automaticamente o Windows após um problema detectado pelo Zabbix, defina o seguinte script:

Parâmetro de script	Valor
Escopo	'Operação de ação'
Tipo	'Script'
Comando	c:\windows\system32\shutdown.exe -r -f

Exemplo 2

Reinicie o host usando o controle IPMI.

Parâmetro de script	Valor
Escopo	'Operação de ação'
Tipo	'IPMI'
Comando	reinicializar

Exemplo 3

Desligue o host usando o controle IPMI.

Parâmetro de script	Valor
Escopo	'Operação de ação'
Tipo	'IPMI'
Comando	desligar

3 Operações adicionais

Visão geral

Nesta seção você pode encontrar alguns detalhes de **adicionais operações** para eventos de descoberta/autoregistro.

Adicionando host

Os hosts são adicionados durante o processo de descoberta, assim que um host é descoberto, e não no final do processo de descoberta.

Como a descoberta de rede pode levar algum tempo devido a muitos hosts/serviços indisponíveis tendo paciência e usando IP razoável intervalos é aconselhável.

Ao adicionar um host, seu nome é decidido pelo padrão **gethostbyname** função. Se o host puder ser resolvido, nome resolvido é usado. Caso contrário, o endereço IP é usado. Além disso, se o endereço IPv6 deve ser usado para um nome de host, então todos os ":" (dois pontos) serão substituídos por "_" (sublinhados), uma vez que os dois pontos não são permitidos em nomes de host.

::: não importante Se estiver realizando a descoberta por um proxy, atualmente A pesquisa do nome do host ainda ocorre no servidor Zabbix. :::

::: não importante Se já existe um host na configuração do Zabbix com o mesmo nome de um recém-descoberto, versões do Zabbix anteriores para 1.8 adicionaria outro host com o mesmo nome. Zabbix 1.8.1 e posterior adiciona **_N** ao nome do host, onde **N** é um número crescente, começando com 2. :::

4 Usando macros em mensagens

Visão geral

Nos assuntos da mensagem e no texto da mensagem, você pode usar macros para mais relatórios de problemas eficientes.

Uma **lista completa de macros** suportado pelo Zabbix está disponível.

Exemplos

Exemplos aqui ilustram como você pode usar macros em mensagens.

Exemplo 1

Assunto da mensagem:

Problema: {TRIGGER.NAME}

Quando você receber a mensagem, o assunto da mensagem será substituído por algo como:

Problema: A carga do processador está muito alta no servidor Zabbix

Exemplo 2

Mensagem:

A carga do processador é: last(/zabbix.zabbix.com/system.cpu.load[,avg1])

Quando você receber a mensagem, a mensagem será substituída por algo Como:

A carga do processador é: 1,45

Exemplo 3

Mensagem:

Último valor: last(/{HOST.HOST}/{ITEM.KEY})

MAX por 15 minutos: max(/{HOST.HOST}/{ITEM.KEY},15m)

MIN por 15 minutos: min(/{HOST.HOST}/{ITEM.KEY},15m)

Quando você receber a mensagem, a mensagem será substituída por algo Como:

Último valor: 1,45
MAX por 15 minutos: 2,33
MIN por 15 minutos: 1,01

Exemplo 4

Mensagem:

`http://<server_ip_or_name>/zabbix/tr_events.php?triggerid={TRIGGER.ID}&eventid={EVENT.ID}`

Quando você receber a mensagem, ela conterá um link para o Evento página de detalhes, que fornece informações sobre o evento, seu acionador, e uma lista dos últimos eventos gerados pelo mesmo gatilho.

Exemplo 5

Informando sobre valores de vários hosts em uma expressão de trigger.

Mensagem:

Nome do problema: {TRIGGER.NAME}

Expressão de gatilho: {TRIGGER.EXPRESSION}

1. Valor do item em {HOST.NAME1}: {ITEM.VALUE1} ({ITEM.NAME1})
2. Valor do item em {HOST.NAME2}: {ITEM.VALUE2} ({ITEM.NAME2})

Quando você receber a mensagem, a mensagem será substituída por algo Como:

Nome do problema: a carga do processador está muito alta em um host local

Expressão de gatilho: `last(/Myhost/system.cpu.load[percpu,avg1])>5` ou `last(/Myotherhost/system.cpu.load[pe`

1. Valor do item no Myhost: 0,83 (carga do processador (média de 1 min por núcleo))
2. Valor do item no Myotherhost: 5,125 (carga do processador (média de 1 min por núcleo))

Exemplo 6

Receber detalhes do evento de problema e do evento de recuperação em um **recuperação** mensagem:

Mensagem:

Problema:

ID do evento: {EVENT.ID}

Valor do evento: {EVENT.VALUE}

Status do evento: {EVENT.STATUS}

Horário do evento: {EVENT.TIME}

Data do evento: {EVENT.DATE}

Idade do evento: {EVENT.AGE}

Confirmação de evento: {EVENT.ACK.STATUS}

Histórico de atualizações de eventos: {EVENT.UPDATE.HISTORY}

Recuperação:

ID do evento: {EVENT.RECOVERY.ID}

Valor do evento: {EVENT.RECOVERY.VALUE}

Status do evento: {EVENT.RECOVERY.STATUS}

Hora do evento: {EVENT.RECOVERY.TIME}

Data do evento: {EVENT.RECOVERY.DATE}

Dados operacionais: {EVENT.OPDATA}

Ao receber a mensagem, as macros serão substituídas por algo Como:

Problema:

ID do evento: 21874

Valor do evento: 1

Status do evento: PROBLEMA

Horário do evento: 13:04:30

Data do evento: 2018.01.02

Idade do evento: 5m

Reconhecimento de evento: Sim

Histórico de atualização do evento: 2018.01.02 13:05:51 "John Smith (Admin)"

Ações: reconhecidas.

Recuperação:

ID do evento: 21896

Valor do evento: 0

Status do evento: OK

Horário do evento: 13:10:07

Data do evento: 2018.01.02

Dados operacionais: 0 valor atual é 0,83

::: não importante Macros de notificação separadas para o original evento de problema e evento de recuperação são suportados desde o Zabbix 2.2.0. :::

3 Operações de recuperação

Visão geral

As operações de recuperação permitem que você seja notificado quando os problemas forem resolvidos.

Tanto as mensagens quanto os comandos remotos são suportados nas operações de recuperação. Embora várias operações possam ser adicionadas, o escalonamento não é suportado - todas operações são atribuídas a uma única etapa e, portanto, serão executadas simultaneamente.

Casos de uso

Alguns casos de uso para operações de recuperação são os seguintes:

1. Notifique todos os usuários que foram notificados sobre o problema
 - * Selecione 'Enviar mensagem de recuperação' como tipo de operação
- Tenha várias operações na recuperação: envie uma notificação e execute um comando remoto
 - * Adicione tipos de operação para enviar uma mensagem e executar um comando
- Abra um ticket no helpdesk externo/sistema de tickets e feche-o quando o problema for resolvido
 - * Crie um script externo que se comunique com o sistema de helpdesk
 - * Crie uma ação tendo operação que execute este script e assim abra um ticket
 - * Tenha uma operação de recuperação que execute este script com outros parâmetros e feche o ticket
 - * Use a macro {EVENT.ID} para fazer referência ao problema original

Configurando uma operação de recuperação

Para configurar uma operação de recuperação, vá para a guia Operações em **action** configuração.

Action
Operations 2

* Default operation step duration

Pause operations for suppressed problems
☒

Operations

Steps
Details

1
Send message to user groups: Zabbix administrators vi

Add

Recovery operations

Details

Notify all involved

Add

Action
Edit

Update operations

Details

Add

Action

* At least one operation must exist.

Para configurar os detalhes de uma nova operação de recuperação, clique em [Add](#) na Recuperação bloco de operações. Para editar uma operação existente, clique em [Edit](#) ao lado do Operação. Uma janela popup será aberta onde você poderá editar a operação detalhes do passo.

Detalhes da operação de recuperação

Operation details

Operation
Send message

* At least one user or user group must be selected.

Send to user groups

User group
Zabbix administrators

Add

Action
Remove

Send to users

User

Add

Action

Send only to
Email

Custom message
☐

Add
Cancel

Parâmetro	Descrição
Operação	<p>Três tipos de operação estão disponíveis para eventos de recuperação:</p> <p>Enviar mensagem - enviar mensagem de recuperação ao usuário especificado</p> <p>Notificar todos os envolvidos - enviar mensagem de recuperação para todos os usuários que foram notificados sobre o evento de problema</p> <p><nome do comando remoto> - execute um comando remoto. Os comandos estão disponíveis para execução se previamente definidos em scripts globais com Operação de ação selecionada como seu escopo.</p> <p>Observe que se o mesmo destinatário for definido em vários tipos de operação sem Mensagem personalizada especificada, notificações duplicadas não são enviadas.</p>
Tipo de operação: enviar mensagem	
Enviar para grupos de usuários	<p>Clique em Adicionar para selecionar grupos de usuários para os quais enviar a mensagem de recuperação.</p> <p>O grupo de usuários deve ter pelo menos permissões de leitura ao host para ser notificado.</p>
Enviar aos usuários	<p>Clique em Adicionar para selecionar os usuários para os quais enviar a mensagem de recuperação.</p> <p>O usuário deve ter pelo menos permissões de leitura ao host para ser notificado.</p>
Enviar apenas para Mensagem personalizada	<p>Enviar mensagem de recuperação padrão para todos os tipos de mídia definidos ou apenas para um selecionado.</p>
Assunto	<p>Se selecionado, uma mensagem personalizada pode ser definida.</p>
Mensagem	<p>Assunto da mensagem personalizada. O assunto pode conter macros.</p> <p>A mensagem personalizada. A mensagem pode conter macros.</p>
Tipo de operação: comando remoto	

Parâmetro		Descrição
	Lista de alvos	<p>Selecione os alvos para executar o comando em:</p> <p>Host atual - o comando é executado no host do gatilho que causou o evento de problema. Esta opção não funcionará se houver vários hosts no gatilho.</p> <p>Host - selecione host(s) para executar o comando.</p> <p>Host group - selecione host group(s) para executar o comando. Especificar um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados. Assim, o comando remoto também será executado em hosts de grupos aninhados.</p> <p>Um comando em um host é executado apenas uma vez, mesmo se o host corresponder mais de uma vez (por exemplo, de vários grupos de hosts; individualmente e de um grupo de hosts).</p> <p>A lista de alvos não tem sentido se o comando for executado no servidor Zabbix. A seleção de mais destinos neste caso resulta apenas na execução do comando no servidor mais vezes.</p> <p>Observe que, para scripts globais, a seleção de destino também depende da configuração Host group no script global configuration.</p>
	Tipo de operação: notificar todos os envolvidos	
	Mensagem personalizada	Se selecionado, uma mensagem personalizada pode ser definida.
	Assunto	Assunto da mensagem personalizada. O assunto pode conter macros.
	Mensagem	A mensagem personalizada. A mensagem pode conter macros.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho. Quando feito, clique em Adicionar para adicionar a operação à lista de Operações de recuperação.

Operation type: **send message**

Parameter	Description
Send to user groups	Click on Add to select user groups to send the recovery message to. The user group must have at least "read" permissions to the host in order to be notified.
Send to users	Click on Add to select users to send the recovery message to. The user must have at least "read" permissions to the host in order to be notified.
Send only to Custom message	Send default recovery message to all defined media types or a selected one only. If selected, a custom message can be defined.
Subject	Subject of the custom message. The subject may contain macros.
Message	The custom message. The message may contain macros.

Operation type: **remote command**

Parameter	Description
Target list	<p>Select targets to execute the command on:</p> <p>Current host - command is executed on the host of the trigger that caused the problem event. This option will not work if there are multiple hosts in the trigger.</p> <p>Host - select host(s) to execute the command on.</p> <p>Host group - select host group(s) to execute the command on. Specifying a parent host group implicitly selects all nested host groups. Thus the remote command will also be executed on hosts from nested groups.</p> <p>A command on a host is executed only once, even if the host matches more than once (e.g. from several host groups; individually and from a host group).</p> <p>The target list is meaningless if the command is executed on Zabbix server. Selecting more targets in this case only results in the command being executed on the server more times.</p> <p>Note that for global scripts, the target selection also depends on the Host group setting in global script configuration.</p>

Operation type: notify all involved

Parameter	Description
Custom message	If selected, a custom message can be defined.
Subject	Subject of the custom message. The subject may contain macros.
Message	The custom message. The message may contain macros.

4 Operações de atualização

Visão geral

As operações de atualização estão disponíveis em ações com o seguinte evento fontes:

- Triggers - quando os problemas são **atualizado** por outros usuários, ou seja, comentado, reconhecido, a gravidade foi alterada, fechada (manualmente);
- Serviços - quando a gravidade de um serviço mudou, mas o serviço ainda não foi recuperado.

Tanto as mensagens quanto os comandos remotos são suportados nas operações de atualização. Embora várias operações possam ser adicionadas, o escalonamento não é suportado - todas operações são atribuídas a uma única etapa e, portanto, serão executadas simultaneamente.

Configurando uma operação de atualização

Para configurar uma operação de atualização, acesse a guia Operações em ação [configuração](#).

Action
Operations 2

* Default operation step duration

Pause operations for suppressed problems
☒

Operations	Steps	Details	Start in	Duration
	Add			

Recovery operations	Details	Action
	Add	

Update operations	Details
	<p>Notify all involved</p> <p>Send message to user groups: Zabbix administrators via SMS</p> <p>Add</p>

Para configurar os detalhes de uma nova operação de atualização, clique em [Add](#) na atualização bloco de operações. Para editar uma operação existente, clique em [Edit](#) ao lado do Operação. Uma janela popup será aberta onde você poderá editar a operação detalhes do passo.

Update operations offer the same set of parameters as **Recovery operations**.

Operation details

Operation

* At least one user or user group must be selected.

Send to user groups	User group	Action
	Zabbix administrators	Remove
	Add	

Send to users	User	Action
	Add	

Send only to

Custom message
☐

Add

Cancel

5 Escalações

Visão geral

Com os escalonamentos, você pode criar cenários personalizados para envio notificações ou execução de comandos remotos.

Em termos práticos significa que:

- Os usuários podem ser informados sobre novos problemas imediatamente
- As notificações podem ser repetidas até que o problema seja resolvido
- O envio de uma notificação pode ser atrasado
- As notificações podem ser escaladas para outro grupo de usuários "superior"
- Comandos remotos podem ser executados imediatamente ou quando um problema não é resolvido por um longo período

As ações são escalonadas com base na **etapa de escalonamento**. Cada passo tem uma duração no tempo.

Você pode definir a duração padrão e uma duração personalizada de um passo individual. A duração mínima de uma etapa de escalonamento é de 60 segundos.

Você pode iniciar ações, como enviar notificações ou executar comandos, a partir de qualquer etapa. O primeiro passo é para ações imediatas. Se você quiser atrasar uma ação, você pode atribuí-la a uma etapa posterior. Para cada passo, várias ações podem ser definidas.

O número de etapas de escalonamento não é limitado.

Os escalonamentos são definidos ao **configurar uma operação**. As escalões são suportados apenas para operações problemáticas, não para recuperação.

Aspectos diversos do comportamento de escalação

Vamos considerar o que acontece em diferentes circunstâncias se uma ação contém várias etapas de escalonamento.

Situação	Comportamento
O host em questão entra em manutenção após o envio da notificação inicial do problema	Dependendo da configuração Pausar operações para problemas suprimidos na ação configuração , todas as etapas de escalonamento restantes são executadas com um atraso causado pelo período de manutenção ou sem atraso. Um período de manutenção não cancela as operações.
O período de tempo definido na condição de ação Período de tempo termina após o envio da notificação inicial	Todas as etapas de escalonamento restantes são executadas. A condição Período de tempo não pode interromper as operações; tem efeito em relação a quando as ações são iniciadas/não iniciadas, não as operações.
Um problema começa durante a manutenção e continua (não é resolvido) após o término da manutenção	Dependendo da configuração Pausar operações para problemas suprimidos na ação configuração , todas as etapas de escalonamento são executadas a partir do momento em que a manutenção termina ou imediatamente.
Um problema começa durante uma manutenção sem dados e continua (não é resolvido) após o término da manutenção Diferentes escalões seguem em sucessão e sobreposição	Ele deve aguardar o disparo do gatilho, antes que todas as etapas de escalonamento sejam executadas. A execução de cada nova escalação substitui a escalação anterior, mas pelo menos uma etapa de escalação que é sempre executada na escalação anterior. Esse comportamento é relevante em ações em eventos que são criados com TODA avaliação de problema do gatilho.

Situação	Comportamento
Durante um escalonamento em andamento (como uma mensagem sendo enviada), com base em qualquer tipo de evento: - a ação é desativada Com base no evento acionador: - o acionador é desabilitado - o host ou item está desabilitado Com base no evento interno sobre gatilhos: - o gatilho está desabilitado Com base no evento interno sobre itens/regras de descoberta de baixo nível: - o item está desabilitado - o host está desabilitado	A mensagem em andamento é enviada e mais uma mensagem no escalonamento é enviada. A mensagem de acompanhamento terá o texto de cancelamento no início do corpo da mensagem (NOTA: Escalação cancelada) nomeando o motivo (por exemplo, NOTA: Escalação cancelada: ação '<Nome da ação>' desabilitada). Desta forma o destinatário é informado que o escalonamento foi cancelado e não serão executadas mais etapas. Esta mensagem é enviada para todos que receberam as notificações anteriormente. O motivo do cancelamento também é registrado no arquivo de log do servidor (começando em Debug Level 3=Aviso).
Durante uma escalação em andamento (como uma mensagem sendo enviada), a ação é excluída	Observe que a mensagem Escalação cancelada também é enviado se as operações forem concluídas, mas as operações de recuperação estiverem configuradas e ainda não forem executadas. Não são enviadas mais mensagens. As informações são registradas no arquivo de log do servidor (começando em Debug Level 3=Aviso), por exemplo: escalação cancelada: ação id:334 excluído

Exemplos de escalonamento

Exemplo 1

Enviando uma notificação repetida uma vez a cada 30 minutos (5 vezes no total) para um grupo de 'Administradores do MySQL'. Para configurar:

- na guia Operações, defina a Duração padrão da etapa de operação para '30m' (30 minutos)
- Defina as etapas de escalonamento como De '1' Para '5'
- Selecione o grupo 'MySQL Administrators' como destinatários da mensagem

As notificações serão enviadas às 0:00, 0:30, 1:00, 1:30, 2:00 horas depois o problema começa (a menos, é claro, que o problema seja resolvido mais cedo).

Se o problema for resolvido e uma mensagem de recuperação for configurada, ela ser enviado para aqueles que receberam pelo menos uma mensagem de problema dentro deste cenário de escalada.

Note:

Se o gatilho que gerou uma escalação ativa for desabilitado, o Zabbix envia uma mensagem informativa sobre isso para todos aqueles que já receberam notificações.

Exemplo 2

Enviando uma notificação atrasada sobre um problema de longa data. Para configurar:

- Na guia Operações, defina a Duração da etapa de operação padrão para '10h' (10 horas)
- Defina as etapas de escalonamento como De '2' Para '2'

Action
Operations 1

* Default operation step duration

Pause operations for suppressed problems
☒

Operations

Steps	Details	Start in	Duration
2	Send message to user groups: Managers via SMS	10:00:00	Default
Add			

Uma notificação só será enviada na Etapa 2 do cenário de escalonamento, ou 10 horas após o início do problema.

Você pode personalizar o texto da mensagem para algo como 'O problema é mais de 10 horas'.

Exemplo 3

Escalando o problema para o chefe.

No primeiro exemplo acima configuramos o envio periódico de mensagens aos administradores do MySQL. Neste caso, os administradores receberão quatro mensagens antes que o problema seja escalado para o gerenciador de banco de dados. Observe que o gerente receberá uma mensagem apenas caso o problema não seja reconhecido ainda, supostamente ninguém está trabalhando nisso.

Action
Operations 2

* Default operation step duration

Pause operations for suppressed problems
☒

Operations

Steps	Details	Start in	Duration
1 - 0	Send message to user groups: MySQL administrators via Email	Immediately	Default
5	Send message to users: Database manager (J S) via all media	02:00:00	Default
Add			

Detalhes da Operação 2:

Operation details

Operation type

Send message

Steps

5 - 5

(0 - infinitely)

Step duration

0

(0 - use action default)

At least one user or user group must be selected.

Send to user groups

User group	Action
Add	

Send to users

User	Action
Database manager	Remove
Add	

Send only to

- All -

Custom message

☒

Subject

Unacknowledged problem: {EVENT.NAME}

Message

Problem started at {EVENT.TIME} on {EVENT.DATE}
 Problem name: {EVENT.NAME}
 Host: {HOST.NAME}
 Severity: {EVENT.SEVERITY}

 Original problem ID: {EVENT.ID}
 {TRIGGER.URL}
 {ESC.HISTORY}

Conditions

Label	Name	Action
A	Event is not acknowledged	Remove
Add		

Update

Cancel

Observe o uso da macro {ESC.HISTORY} na mensagem personalizada. A macro conterá informações sobre todos os passos executados anteriormente neste escalação, como notificações enviadas e comandos executados.

Exemplo 4

Um cenário mais complexo. Após várias mensagens para administradores do MySQL e escalação para o gerente, o Zabbix tentará reiniciar o MySQL base de dados. Isso acontecerá se o problema persistir por 2:30 horas e não foi reconhecido.

Se o problema persistir, após mais 30 minutos o Zabbix enviará um mensagem para todos os usuários convidados.

Se isso não ajudar, depois de mais uma hora o Zabbix irá reiniciar o servidor com o banco de dados MySQL (segundo comando remoto) usando comandos IPMI.

Action

Operations 5

* Default operation step duration

30m

Pause operations for suppressed problems

☒

Operations

Steps	Details	Start in	Duration
1 - 0	Send message to user groups: MySQL Administrators via Email	Immediately	Default
5	Send message to users: Database Manager (J S) via all media	02:00:00	Default
6	Run script "Restart MySQL" on current host	02:30:00	Default
7	Send message to user groups: Guests via all media	03:00:00	Default
9	Run script "Restart server" on current host	04:00:00	Default

Add

Exemplo 5

Um escalonamento com várias operações atribuídas a uma etapa e personalizado intervalos utilizados. A duração da etapa de operação padrão é de 30 minutos.

Action

Operations 4

* Default operation step duration

30m

Pause operations for suppressed problems

☒

Operations

Steps	Details	Start in	Duration
1 - 4	Send message to user groups: MySQL Administrators via Email	Immediately	Default
5 - 6	Send message to users: Database Manager (J S) via all media	02:00:00	1h
5 - 7	Send message to user groups: Zabbix administrators via Email	02:00:00	10m
11	Send message to user groups: Guests via Email	04:00:00	Default

Add

As notificações serão enviadas da seguinte forma:

- para administradores do MySQL às 0:00, 0:30, 1:00, 1:30 após o problema começa
- para o gerenciador de banco de dados às 2:00 e 2:10 (e não às 3:00; vendo que os passos 5 e 6 se sobrepõem com a próxima operação, o mais curto a duração da etapa de 10 minutos na próxima operação substitui a duração do passo mais longa de 1 hora tentou definir aqui)
- para administradores do Zabbix às 2:00, 2:10, 2:20 após o problema inicia (a duração da etapa personalizada de 10 minutos de trabalho)
- para usuários convidados às 4:00 horas após o início do problema (o padrão duração do passo de 30 minutos retornando entre os passos 8 e 11)

3 Recebendo notificação sobre itens não suportados

Visão geral

Receber notificações sobre itens não suportados é suportado desde o Zabbix 2.2.

Faz parte do conceito de eventos internos do Zabbix, permitindo aos usuários ser notificado nestas ocasiões. Os eventos internos refletem uma mudança de Estado:

- quando os itens vão de 'normal' para 'sem suporte' (e vice-versa)
- quando os gatilhos passam de 'normal' para 'desconhecido' (e vice-versa)
- quando as regras de descoberta de baixo nível passam de 'normal' para 'sem suporte' (e volta)

Esta seção apresenta instruções para **receber notificação** quando um item não é suportado.

Configuração

No geral, o processo de configuração da notificação deve parecer familiar para aqueles que configuraram alertas no Zabbix antes.

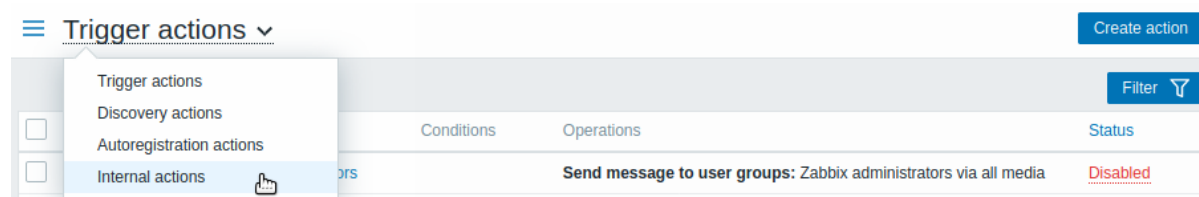
Passo 1

Configure **algumas mídias**, como e-mail, SMS ou script para usar para as notificações. Consulte as seções correspondentes do manual para realizar esta tarefa.

::: não importante Para notificar sobre eventos internos o padrão gravidade ('Não classificado') é usado, então deixe-o marcado quando configurando **usuário media** se quiser para receber notificações de eventos internos. :::

Passo 2

Vá para Configuração → Ações e selecione Ações internas na menu suspenso do título da página.

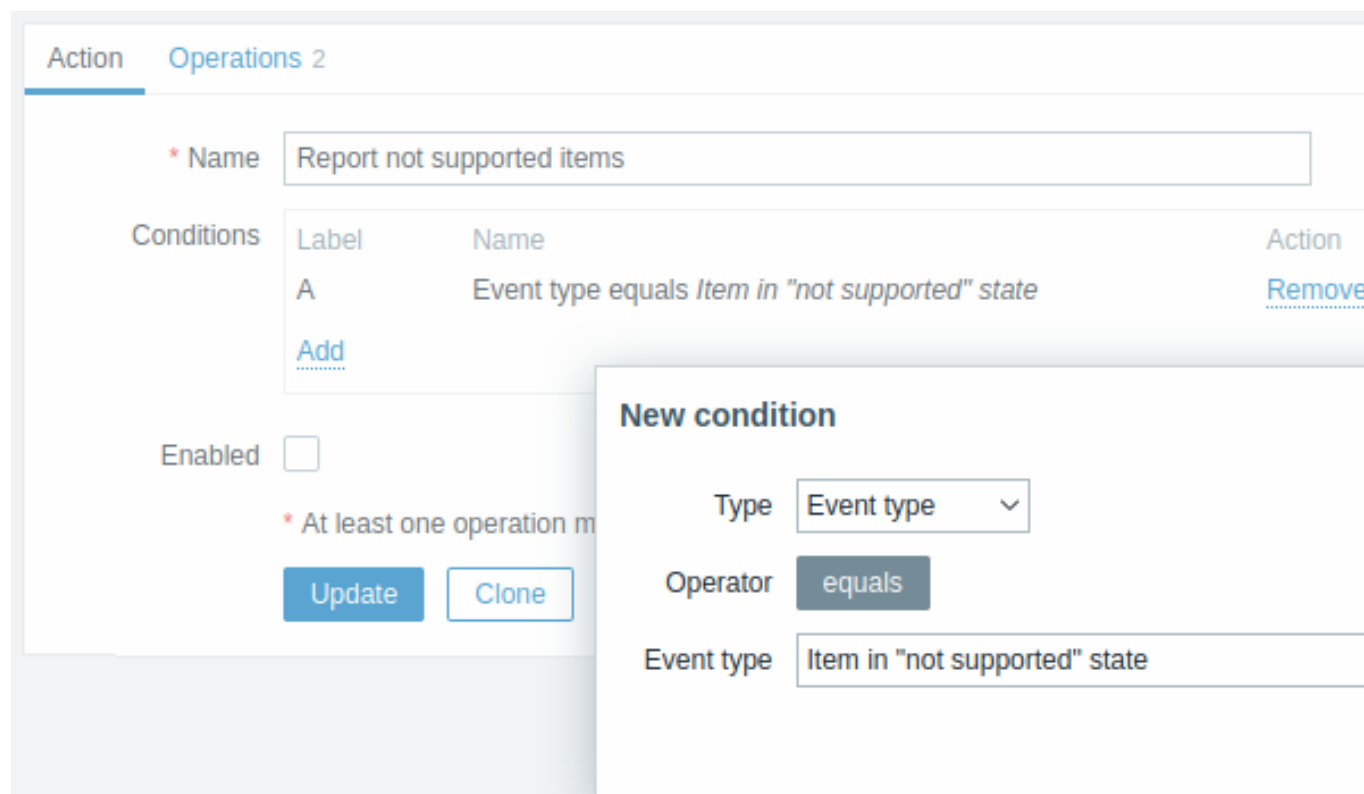


Clique em Criar ação à direita para abrir uma configuração de ação Formato.

Etapas 3

Na guia **Ação**, insira um nome para a ação. Em seguida, clique em Adicionar no bloco de condição para adicionar uma nova condição.

Na nova janela pop-up de condição, selecione Tipo de evento como a condição digite e selecione Item no estado "não suportado" como o tipo de evento valor.



Não se esqueça de clicar em Adicionar para realmente listar a condição no Condições bloco.

Passo 4

Na guia **Operations**, clique em Add no bloco Operations e selecione alguns destinatários da mensagem (grupos de usuários/usuários) e a mídia tipos (ou 'Todos') a serem usados para entrega.

Marque a caixa de seleção Mensagem personalizada se desejar inserir a assunto/conteúdo da mensagem do problema.

* Default operation step duration

Operations

Steps Details

1 **Send message to user groups: Zabbix administrators via all media**

[Add](#)

Recovery operations

Details

Notify all involved

[Add](#)

Action

[Edit](#) [Remove](#)

Operation details

Operation type Send message

Steps - (0 - infinitely)

Step duration (0 - use action default)

* At least one user or user group must be selected.

Send to user groups

User group

Zabbix administrators

[Add](#)

Action

[Remove](#)

Send to users

User

[Add](#)

Action

Send only to

Custom message ☒

Subject

Message

Host: {HOST.NAME}
Item: {ITEM.NAME}
Key: {ITEM.KEY}
State: {ITEM.STATE}

Clique em Add para realmente listar a operação no bloco Operations.

Se você deseja receber mais de uma notificação, defina a operação duração do passo (intervalo entre as mensagens enviadas) e adicionar outro passo.

Etapa 5

O bloco **Operações de recuperação** permite configurar uma recuperação notificação quando um item volta ao estado normal. Clique em Adicionar no bloco Operações de recuperação, selecione o tipo de operação, o destinatários da mensagem (grupos de usuários/usuários) e os tipos de mídia (ou 'Todos') para usar para entrega.

Marque a caixa de seleção Mensagem personalizada se desejar inserir o assunto/conteúdo da mensagem do problema.

Clique em Adicionar na janela pop-up Detalhes da operação para realmente listar a operação no bloco Operações de recuperação.

Etapa 6

Quando terminar, clique no botão **Adicionar** na parte inferior do formulário.

E pronto, pronto! Agora você pode esperar receber seu primeira notificação do Zabbix se algum item não for suportado.

11 Macros

Visão geral

O Zabbix suporta uma série de macros embutidas que podem ser usadas em várias situações. Essas macros são variáveis, identificadas por uma sintaxe específica:

{MACRO}

As macros resolvem para um valor específico dependendo do contexto.

O uso eficaz de macros permite economizar tempo e tornar o Zabbix configuração mais transparente.

Em um dos usos típicos, uma macro pode ser usada em um modelo. Assim um gatilho em um modelo pode ser chamado de "A carga do processador está muito alta no {HOST.NAME}". Quando o modelo é aplicado ao host, como Zabbix servidor, o nome será resolvido para "A carga do processador está muito alta no Zabbix server" quando o gatilho é exibido na seção Monitoring.

As macros podem ser usadas nos parâmetros-chave do item. Uma macro pode ser usada apenas para uma parte do parâmetro, por exemplo `item.key[server_{HOST.HOST}_local]`. A citação dupla do parâmetro é não é necessário, pois o Zabbix cuidará de quaisquer símbolos especiais ambíguos, se presente na macro resolvida.

Além de macros embutidas, o Zabbix também suporta macros definidas pelo usuário, macros definidas pelo usuário com contexto e macros para descoberta de baixo nível.

Veja também:

- lista completa de **incorporado macros**

- macro **funções**
- **macros do usuário**
- **macros de usuário com context**
- **macros de descoberta de baixo nível**

1 Funções macro

Visão geral

As funções de macro oferecem a capacidade de personalizar valores de **macro**.

Às vezes, uma macro pode resolver para um valor que não é necessariamente fácil de trabalhar com. Pode ser longo ou conter uma substring específica de interesse que você gostaria de extrair. É aqui que as funções macro podem ser útil.

A sintaxe de uma função macro é:

```
{<macro>.<func>(<params>)}
```

Onde:

- **<macro>** - a macro para personalizar (por exemplo {ITEM.VALUE} ou {#LLDMACRO})
- **<func>** - a função a ser aplicada
- **<params>** - uma lista delimitada por vírgulas de parâmetros de função. Os parâmetros devem ser citados se começarem com " " (espaço), " ou contém), , .

Por exemplo:

```
{{TIME}.fmttime(format,time_shift)}
{{ITEM.VALUE}.regsub(padrao, saída)}
{{#LLDMACRO}.regsub(padrao, saída)}
```

Funções de macro suportadas

FUNÇÃO	Descrição	Parâmetros	Suportado para
fmtnum (<dígitos>)	Formatação numérica para controlar o número de dígitos após o ponto decimal. Nenhum dos pontos após o ponto decimal.	dígitos - o número de dígitos após o ponto decimal. Nenhum zero à direita será produzido.	{ITEM.VALUE} {ITEM.LASTVALUE} Macros de expressão
fmttime (<format>,<time_shift>)			

Formatação **format** - {TIME}
de hora. string de
formato
obri-
gatório,
com-
patível
com a
for-
matação
da
função
strftime
time_shift
- a mu-
dança
de hora
aplicada
à hora
antes da
for-
matação;
deve
começar
com
-<N><time_unit>
ou
+<N><time_unit>,
onde
N - o
número
de
unidades
de
tempo
para adi-
cionar
ou
subtrair;
time_unit
- h
(hora), d
(dia), w
(sem-
ana), M
(mês) ou
y (ano).
Desde o
Zab-
bix 5.4, o
parâmetro
time_shift
suporta
oper-
ações de
tempo
de
várias
etapas e
pode
incluir /
<time_unit>
para
mudar
para o

FUNÇÃO

iregsub (<padrão>,<saída>)

Extração de sub-string por uma correspondência de expressão regular (não diferencia maiúsculas de minúsculas).	pattern - a expressão regular para corresponder às opções de saída. \1 - \9 marcadores de posição são suportados para capturar grupos. \0 retorna o texto correspondente.	{ITEM.VALUE} {ITEM.LASTVALUE} Macros de descoberta de baixo nível (exceto em filtro de regra de descoberta de nível de nível)
--	---	---

regsub (<padrão>,<saída>)

Extração de sub-string por uma correspondência de expressão regular (diferencia maiúsculas de minúsculas).	padrão - a expressão regular para corresponder às opções de saída. \1 - \9 marcadores de posição são suportados para capturar grupos. \0 retorna o texto correspondente.	{ITEM.VALUE} {ITEM.LASTVALUE} Macros de descoberta de baixo nível (exceto em filtro de regra de descoberta de nível de nível)
--	--	---

Se uma função for usada em um **suportado location**, mas aplicado a uma macro que não suporta funções de macro, a macro é avaliada como 'DESCONHECIDO'.

Se o padrão não for uma expressão regular correta, a macro avalia para 'UNKNOWN' (excluindo macros de descoberta de baixo

nível onde a função será ignorado nesse caso e a macro permanecerá não expandida)

Se uma função de macro for aplicada à macro em locais que não suportam funções de macro, então a função é ignorada.

Exemplos

As maneiras pelas quais as funções de macro podem ser usadas para personalizar valores de macro é ilustrado nos seguintes exemplos de valores recebidos:

Valor recebido	Macro	Saída
24.3413523	{{ITEM.VALUE}.fmtnum(2,34)	24.34
24.3413523	{{ITEM.VALUE}.fmtnum(20)}	24
12:36:01	{{TIME}.fmttime(%B)}	Outubro
12:36:01	{{TIME}.fmttime(%d 1 de setembro %B,-1M/M)}	1 de setembro
123Linha de log	{{ITEM.VALUE}.regsub('^(0+)+', Problem))	123
123 Log line	{{ITEM.VALUE}.regsub('^(0+)+', "Problem"))	123
123 Log line	{{ITEM.VALUE}.regsub('ID do problema: 123 ID do problema: \1)}	123
Linha de log	{{ITEM.VALUE}.regsub('ID do problema: "ID do problema: \1)}	123
MySQL travou erro 123	{{ITEM.VALUE}.regsub('ID do problema: ([0-9]+)', "ID do problema: MySQL_123 " \1_ \2 ")} {{ITEM.VALUE}.regsub('UNIQUE' * expressão "ID do problema: regular inválida) \1)}	123
123 Log line	{{ITEM.VALUE}.regsub('UNIQUE' * expressão "ID do problema: regular inválida) \1)}	123
customernome_1	{{#IFALIAS}.regsub("customernome", \1)}	123
customernome_1	{{#IFALIAS}.regsub("1.*")_([0-9]+)", \2)}	123
customernome_1	{{#IFALIAS}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) (expressão regular inválida)	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) cliente"	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("1.*")_([0-9]+)", \2)"	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) {{#M}.regsub("(.*)_([0-9]+)", \1) (expressão regular inválida)	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) cliente"	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("1.*")_([0-9]+)", \2)"	123
customernome_1	{{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) {{#IFALIAS}}.regsub("{{#IFALIAS}}.regsub("(.*)_([0-9]+)", \1) (expressão regular inválida)	123

2 Macros de usuário

Visão geral

Macros de usuário são suportadas no Zabbix para maior flexibilidade, além de para as macros suportadas sai da caixa.

As macros de usuário podem ser definidas em nível global, de modelo e de host. Esses macros têm uma sintaxe especial:

{{MACRO}}

O Zabbix resolve macros de acordo com a seguinte precedência:

1. macros de nível de host (verificadas primeiro)
2. macros definidas para modelos de primeiro nível do host (ou seja, modelos vinculados diretamente ao host), classificadas por ID de modelo
3. macros definidas para modelos de segundo nível do host, classificadas por ID do modelo
4. macros definidas para modelos de terceiro nível do host, classificadas por ID do modelo, etc.
5. macros globais (marcadas por último)

Em outras palavras, se uma macro não existir para um host, o Zabbix tentará encontrá-lo nos modelos de host de profundidade crescente. Se ainda não for encontrado, um macro global será usada, se existir.

Se uma macro com o **mesmo nome** existir em vários modelos vinculados do mesmo nível, a macro do modelo com o ID mais baixo será usado. Assim tendo macros com o mesmo nome em vários modelos é um risco de configuração.

Se o Zabbix não conseguir encontrar uma macro, a macro não será resolvida.

::: não importante Macros (incluindo macros de usuário) são deixadas sem solução na seção Configuração (por exemplo, na lista de gatilhos) por design para tornar a configuração complexa mais transparente. :::

As macros do usuário podem ser usadas em:

- parâmetro chave do item
- intervalos de atualização de itens e intervalos flexíveis
- nome e descrição do gatilho
- acionar parâmetros e constantes de expressão (consulte [exemplos](#))
- muitos outros locais - veja o [completo lista](#)

Common use cases of global and host macros

- use a global macro in several locations; then change the macro value and apply configuration changes to all locations with one click
- take advantage of templates with host-specific attributes: passwords, port numbers, file names, regular expressions, etc.

Configuration





To define user macros, go to the corresponding location in the frontend:

- for global macros, visit Administration → General → Macros
- for host and template level macros, open host or template properties and look for the Macros tab

Note:






If a user macro is used in items or triggers in a template, it is suggested to add that macro to the template even if it is defined on a global level. That way, if the macro type is text exporting the template to XML and importing it in another system will still allow it to work as expected. Values of secret macros are not **exported**.

A user macro has the following attributes:

Macro	Value		Description
<input data-bbox="204 1473 636 1507" type="text" value="{MYSQL_PASSWORD}"/>	<input data-bbox="651 1473 1099 1507" type="password" value="*****"/>		<input data-bbox="1185 1473 1313 1507" type="text" value="description"/>
<input data-bbox="204 1529 636 1563" type="text" value="{MYSQL_USERNAME}"/>	<input data-bbox="651 1529 1099 1563" type="password" value="*****"/>		<input data-bbox="1185 1529 1313 1563" type="text" value="description"/>
<input data-bbox="204 1585 636 1619" type="text" value="{SECRET_PASSWORD}"/>	<input data-bbox="651 1585 1099 1619" type="text" value="path/to/secret:password"/>		<input data-bbox="1185 1585 1313 1619" type="text" value="description"/>
<input data-bbox="204 1641 636 1675" type="text" value="{SECRET_USERNAME}"/>	<input data-bbox="651 1641 1099 1675" type="text" value="path/to/secret:username"/>		
<input data-bbox="204 1697 636 1731" type="text" value="{SNMP_COMMUNITY}"/>	<input data-bbox="651 1697 1099 1731" type="text" value="public"/>		
<input data-bbox="204 1753 636 1787" type="text" value="{WORKING_HOURS}"/>	<input data-bbox="651 1753 1099 1787" type="text" value="1-5,09:00-18:00"/>		<input data-bbox="1185 1753 1313 1787" type="text" value="description"/>

[Add](#)

Parameter	Description
Macro	Macro name. The name must be wrapped in curly brackets and start with a dollar sign. Example: {\$FRONTEND_URL}. The following characters are allowed in the macro names: A-Z (uppercase only) , 0-9 , _ , .

Parameter	Description
Value	<p>Macro value. Three value types are supported:</p> <p>Text (default) - plain-text value</p> <p>Secret text - the value is masked with asterisks, which could be useful to protect sensitive information such as passwords or shared keys.</p> <p>Vault secret - the value contains a reference path (as 'path:key', for example "secret/zabbix:password") to a Vault secret</p> <p>Note that while the value of a secret macro is hidden from sight, the value can be revealed through the use in items. For example, in an external script an 'echo' statement referencing a secret macro may be used to reveal the macro value to the frontend because Zabbix server has access to the real macro value.</p> <p>To select the value type click on the button at the end of the value input field:</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 5px;">T ▼</div> <div>icon indicates a text macro;</div> </div> <div style="display: flex; align-items: center;"> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 5px;"> ▼</div> <div>icon indicates</div> </div> <p>a secret text macro. Upon hovering, the value field transforms into a</p> <div style="background-color: #0070c0; color: white; padding: 5px; display: flex; justify-content: space-between; align-items: center;"> Set new value   </div> <p>button, which allows to enter a new value of the macro (to exit without saving a new value, click the backwards arrow ().</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid #ccc; padding: 2px; margin-right: 5px;"> ▼</div> <div>icon indicates a secret Vault macro.</div> </div> <p>Maximum length of a user macro value is 2048 characters (255 characters in versions before 5.2.0).</p>
Description	Text field used to provide more information about this macro.

Note:

URLs that contain a secret macro will not work as the macro in them will be resolved as "*****".

Attention:

In trigger expressions user macros will resolve if referencing a parameter or constant. They will NOT resolve if referencing a host, item key, function, operator or another trigger expression. Secret macros cannot be used in trigger expressions.

Examples

Example 1

Use of host-level macro in the "Status of SSH daemon" item key:

```
net.tcp.service[ssh,,{${SSH_PORT}]
```

This item can be assigned to multiple hosts, providing that the value of **{\${SSH_PORT}}** is defined on those hosts.

Example 2

Use of host-level macro in the "CPU load is too high" trigger:

```
last(/ca_001/system.cpu.load[,avg1])>{${MAX_CPULOAD}}
```

Such a trigger would be created on the template, not edited in individual hosts.

Note:

If you want to use the amount of values as the function parameter (for example, **max(/host/key,#3)**), include hash mark in the macro definition like this: **SOME_PERIOD => #3**

Example 3

Use of two macros in the "CPU load is too high" trigger:

```
min(/ca_001/system.cpu.load[,avg1],{$CPULOAD_PERIOD})>{$MAX_CPULOAD}
```

Note that a macro can be used as a parameter of trigger function, in this example function **min()**.

Example 4

Synchronize the agent unavailability condition with the item update interval:

- define {\$INTERVAL} macro and use it in the item update interval;
- use {\$INTERVAL} as parameter of the agent unavailability trigger:

```
nodata(/ca_001/agent.ping,{$INTERVAL})=1
```

Example 5

Centralize configuration of working hours:

- create a global {\$WORKING_HOURS} macro equal to 1-5,09:00-18:00;
- use it in the Working time field in Administration → General → GUI;
- use it in the When active field in Administration → User → Media;
- use it to set up more frequent item polling during working hours:

Update interval

{\$LONG_INTERVAL}

Custom intervals

Type	Interval	Period
Flexible	Scheduling	<div>{\$SHORT_INTERVAL}</div> <div>{\$WORKING_HOURS}</div>

- use it in the Time period action condition;
- adjust the working time in Administration → General → Macros, if needed.

Example 6

Use host prototype macro to configure items for discovered hosts:

- on a host prototype define user macro {\$SNMPVALUE} with {#SNMPVALUE} **low-level discovery** macro as a value:

Host prototype macros

Inherited and host prototype macros

Macro	Value
{\$SNMPVALUE}	<div>{#SNMPVALUE}</div> <div>T</div>

Add

Cancel

- assign Generic SNMPv2 template to the host prototype;
- use {\$SNMPVALUE} in the SNMP OID field of Generic SNMPv2 template items.

User macro context

See **user macros with context**.

3 Macros de usuário com contexto

Visão geral

Um contexto opcional pode ser usado em **user macros**, permitindo substituir o valor padrão com um específico do contexto.

O contexto é anexado ao nome da macro; a sintaxe depende se o contexto é um valor de texto estático:

```
{$MACRO:"texto estático"}
```

ou uma expressão regular:

```
{${MACRO:regex:"expressão regular"}}
```

Observe que uma macro com contexto de expressão regular só pode ser definida em configuração de macro do usuário. Se o prefixo `regex:` for usado em outro lugar como contexto de macro do usuário, como em uma expressão de gatilho, será tratado como contexto estático.

A citação de contexto é opcional (veja também [importante notas](#)).

Exemplos de contexto de macro:

Exemplo	Descrição
<code>{\${LOW_SPACE_LIMIT}}</code>	Macro do usuário sem contexto.
<code>{\${LOW_SPACE_LIMIT:/tmp}}</code>	Macro do usuário com contexto (string estática).
<code>{\${LOW_SPACE_LIMIT:regex:"~/tmp\$"}}</code>	Macro de usuário com contexto (expressão regular). O mesmo que <code>{\${LOW_SPACE_LIMIT:/tmp}}</code> .
<code>{\${LOW_SPACE_LIMIT:regex:"~/var/log/.*\$"}}</code>	Macro de usuário com contexto (expressão regular). Corresponde a todas as strings prefixadas com <code>/var/log/</code> .

Casos de uso

Macros de usuário com contexto podem ser definidas para realizar limites em expressões de gatilho (com base nos valores recuperados por descoberta de baixo nível). Por exemplo, você pode definir as seguintes macros:

- `{${LOW_SPACE_LIMIT}} = 10`
- `{${LOW_SPACE_LIMIT:/home}} = 20`
- `{${LOW_SPACE_LIMIT:regex:"^\\[a-z]+$"} = 30`

Então, uma macro de descoberta de baixo nível pode ser usada como contexto de macro em um acionar protótipo para descoberta do sistema de arquivos montado:

```
last(/host/vfs.fs.size[{#FSNAME},pfree])<{${LOW_SPACE_LIMIT:"{#FSNAME}"}
```

Após a descoberta, diferentes limites de espaço reduzido serão aplicados em acionadores dependendo dos pontos de montagem ou tipos de sistema de arquivos descobertos. Os eventos de problema serão gerados se:

- A pasta `/home` tem menos de 20% de espaço livre em disco
- pastas que correspondem ao padrão `regex` (como `/etc`, `/tmp` ou `/var`) têm menos de 30% do espaço livre em disco
- pastas que não correspondem ao padrão `regex` e não são `/home` têm menos de 10% do espaço livre em disco

Anotações importantes

- Se existir mais de uma macro de usuário com contexto, o Zabbix tentará combinar as macros de contexto simples primeiro e depois as macros de contexto com expressões regulares em uma ordem indefinida.

Não crie macros de contexto diferentes que correspondam ao mesma string para evitar comportamento indefinido.

- Se uma macro com seu contexto não for encontrada no host, modelos vinculados ou globalmente, a macro sem contexto é pesquisada.
- Somente macros de descoberta de baixo nível são suportadas no contexto. Algumas outras macros são ignoradas e tratadas como texto simples.

Tecnicamente, o contexto macro é especificado usando regras semelhantes a [item key](#), exceto que o contexto da macro é não analisado como vários parâmetros se houver um caractere `,`:

- O contexto da macro deve ser citado com `"` se o contexto contiver um `}` caractere ou começa com um caractere `"`. Aspas entre aspas contexto deve ser escapado com o caractere `\\`.
- O caractere `\\` em si não tem escape, o que significa que é impossível ter um contexto citado terminando com o caractere `\\` - a macro `{${MACRO:"a:\\b\\c"}}` é inválido.
- Os espaços iniciais no contexto são ignorados, os espaços finais são não:
 - Por exemplo, `{${MACRO:A}}` é igual a `{${MACRO: A}}`, mas não `{${MACRO:A }}`.
- Todos os espaços antes das cotas iniciais e após as cotas finais são ignorado, mas todos os espaços entre aspas não são:
 - Macros `{${MACRO:"A"}}`, `{${MACRO: "A"}}`, `{${MACRO:"A" }}` e `{${MACRO: "A" }}` são iguais, mas as macros `{${MACRO:"A" }}` e `{${MACRO:" A "}}` não são.

As macros a seguir são todas equivalentes, pois possuem a mesma contexto: `{${MACRO:A}}`, `{${MACRO: A}}` e `{${MACRO:"A"}}`. Isso está em contraste com chaves de item, onde `'key[a]'`, `'key[a]'` e `'key["a"]'` são as mesmo semanticamente, mas diferente para fins de exclusividade.

4 Macros de descoberta de baixo nível

Visão geral

Existe um tipo de macro usado no **baixo nível descoberta** (LLD) função:

{#MACRO}

É uma macro que é usada em uma regra LLD e retorna valores reais da nome do sistema de arquivos, interface de rede, SNMP OID, etc.

Essas macros podem ser usadas para criar itens, gatilhos e gráficos protótipos. Então, ao descobrir sistemas de arquivos reais, redes interfaces etc., essas macros são substituídas por valores reais e são a base para a criação de itens reais, gatilhos e gráficos.

Essas macros também são usadas na criação de protótipos de host e grupo de host na máquina virtual **discovery**.

Algumas macros de descoberta de baixo nível vêm "pré-empacotadas" com o LLD função no Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. No entanto, a adesão a esses nomes não é obrigatória quando criando um **personalizado** regra de descoberta de baixo nível. Então você pode usar qualquer outro nome de macro LLD e referir a esse nome.

Locais compatíveis

As macros LLD podem ser usadas:

- no filtro de regra de descoberta de baixo nível
- para protótipos de itens em
 - nome
 - parâmetros chave
 - unidade
 - intervalo de atualização¹
 - período de armazenamento do histórico¹
 - período de armazenamento de tendências¹
 - etapas de pré-processamento do valor do item
 - SNMP OID
 - Campo do sensor IPMI
 - fórmula de item calculado
 - Script SSH e script Telnet
 - consulta SQL de monitoramento de banco de dados
 - Campo de ponto final do item JMX
 - Descrição
 - Campo de URL do agente HTTP
 - Campo de campos de consulta HTTP do agente HTTP
 - Campo do corpo da solicitação do agente HTTP
 - Campo de códigos de status obrigatório do agente HTTP
 - Chave e valor do campo de cabeçalhos do agente HTTP
 - Campo de nome de usuário de autenticação HTTP do agente HTTP
 - Campo de senha de autenticação HTTP do agente HTTP
 - Campo de proxy HTTP do agente HTTP
 - Campo de arquivo de certificado HTTP SSL do agente HTTP
 - Campo de arquivo de chave HTTP SSL do agente HTTP
 - Campo de senha da chave HTTP SSL do agente HTTP
 - Tempo limite HTTP do agente HTTP¹ campo
 - Tag
- para protótipos de gatilho em
 - nome
 - Dados operacionais
 - expressão (somente em constantes e parâmetros de função)
 - URL
 - Descrição
 - Tag
- para protótipos de gráficos em
 - nome
- para protótipos de host em
 - nome
 - nome visível
 - campos de interface personalizados: IP, DNS, porta, comunidade SNMP v1/v2, Nome de contexto SNMP v3, nome de segurança SNMP v3, SNMP v3 senha de autenticação, senha de privacidade SNMP v3

- nome do protótipo do grupo de hosts
- valor da tag do host
- valor da macro do host
- (veja a [lista completa](#))

Em todos esses lugares, as macros LLD podem ser usadas dentro do usuário estático **macro context**.

Usando funções de macro

As funções de macro são suportadas com macros de descoberta de baixo nível (exceto em filtro de regras de descoberta de baixo nível), permitindo extrair uma certa parte o valor da macro usando uma expressão regular.

Por exemplo, você pode querer extrair o nome do cliente e a interface número da seguinte macro LLD para fins de marcação de eventos:

```
{#IFALIAS}=nomedocliente_1
```

Para fazer isso, a função macro `regsub` pode ser usada com a macro no campo de valor de tag de evento de um protótipo de acionador:

Tags	Customer	{#{IFALIAS}.regsub("(.*)_([0-9]+)", \1)}	Remove
	Interface	{#{IFALIAS}.regsub("(.*)_([0-9]+)", \2)}	Remove

Observe que as vírgulas não são permitidas no item sem aspas **chave parâmetros**, então o parâmetro contendo uma função macro deve ser citado. A barra invertida (\) deve ser usado para escapar de aspas duplas dentro do parâmetro. Exemplo:

```
net.if.in["{#{IFALIAS}.regsub(\"(.*)_([0-9]+)\", \1)}", bytes]
```

Para obter mais informações sobre a sintaxe da função de macro, consulte: [Macro funções](#)

Funções de macro são suportadas em macros de descoberta de baixo nível desde o Zabbix 4.0.

Notas de rodapé

¹ Nos campos marcados com ¹ um único macro tem que preencher todo o campo. Várias macros em um campo ou macros misturados com texto não são suportados.

5 Macros de expressão

Visão geral

Macros de expressão são úteis para cálculos de formula. Elas são calculadas expandindo todas as macros internas e avaliando a expressão resultante.

As macros de expressão possuem uma sintaxe especial:

```
{?EXPRESSION}
```

A sintaxe na EXPRESSION é a mesma como em [trigger expressions](#) (veja as limitações de uso abaixo).

{HOST.HOST<1-9>} and {ITEM.KEY<1-9>} macros são suportadas dentro de macros de expressão. {ITEM.KEY<1-9>} Macros são suportadas dentro de macros de expressão desde o Zabbix 6.0.9.

Uso

Nas seguintes localizações:

- graph names
- map element labels
- map shape labels
- map link labels

somente uma função **single**, do conjunto a seguir: `avg`, `last`, `max`, `min`, é permitida como uma expressão macro, e.g.:

```
{?avg(/{HOST.HOST}/{ITEM.KEY},1h)}
```

Expressões como `{?last(/host/item1)/last(/host/item2)}`, `{?count(/host/item1,5m)}` e `{?last(/host/item1)*10}` são incorretas nesses locais.

Contudo, em:

- trigger event names
- trigger-based notifications and commands
- problem update notifications and commands

As expressões **complex** são permitidas, e.g.:

```
{?trendavg(/host/item1,1M:now/M)/trendavg(/host/item1,1M:now/M-1y)*100}
```

Ver também:

- **Supported macros**,
- para a lista de locais suportados da macro expressão
- **Example** de uso de uma macro expressão no event name

12 Usuários e grupos de usuários

Visão geral

Todos os usuários do Zabbix acessam o aplicativo Zabbix através do frontend web. Cada usuário recebe um nome de login exclusivo e uma senha.

Todas as senhas dos usuários são criptografadas e armazenadas no banco de dados Zabbix. Os usuários não podem usar seu ID de usuário e senha para fazer login diretamente no servidor UNIX, a menos que também tenham sido configurados dessa forma no UNIX. A comunicação entre o servidor web e o navegador do usuário pode ser protegida usando SSL.

Com um flexível **esquema de permissão de usuário** você pode restringir e diferenciar os direitos de:

- acessar funções administrativas de frontend do Zabbix
- executar certas ações no frontend
- acessar hosts monitorados em grupos de hosts
- use métodos de API específicos

1 Configurando um usuário

Visão geral

A instalação inicial do Zabbix tem dois usuários predefinidos:

- Admin - um **superusuário** Zabbix com permissões totais;
- convidado - um **usuário** Zabbix especial. O usuário 'convidado' é desabilitado por padrão. Se você adicioná-lo ao grupo de usuário Guests, você pode acessar as páginas de monitoramento no Zabbix sem estar conectado. Observe que, por padrão, 'convidado' não tem permissões em Objetos Zabbix.

Para configurar um novo usuário:

- Vá para Administração → Usuários
- Clique em Criar usuário (ou no nome do usuário para editar um usuário existente)
- Editar atributos do usuário no formulário

Atributos gerais

A guia Usuário contém atributos gerais do usuário:

User
Media 1
Permissions

* Username

Admin

Name

Zabbix

Last name

Administrator

* Groups

Zabbix administrators

type here to search

Select

Password

Change password

Language

English (en_US)

Time zone

System default: (UTC+02:00) Europe/Riga

Theme

Blue

Auto-login

☒

Auto-logout

☐

15m

* Refresh

30s

* Rows per page

50

URL (after login)

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome de usuário	Nome de usuário exclusivo, usado como nome de login.
Nome	Nome do usuário (opcional). Se não estiver vazio, visível nas informações de confirmação e nas informações do destinatário da notificação.
Sobrenome	Sobrenome do usuário (opcional). Se não estiver vazio, visível nas informações de confirmação e nas informações do destinatário da notificação.
Grupos	Selecione grupos de usuários ao qual o usuário pertence. A partir do Zabbix 3.4.3 este campo é auto-preenchido, então começar a digitar o nome de um grupo de usuários oferecerá uma lista suspensa de grupos correspondentes. Role para baixo para selecionar. Alternativamente, clique em Selecionar para adicionar grupos. Clique em 'x' para remover o selecionado. A adesão aos grupos de usuários determina a quais grupos de hosts e hosts o usuário terá acesso .
Senha	Dois campos para inserir a senha do usuário. Com uma senha existente, contém um botão Senha, clicando nele abre os campos de senha.
Idioma	Observe que senhas com mais de 72 caracteres serão truncadas. Idioma do frontend Zabbix. A extensão php gettext é necessária para que as traduções funcionem.

Parâmetro	Descrição
Fuso horário	Selecione o fuso horário para substituir o [fuso horário] global (/manual/web_interface/time_zone#overview) no nível do usuário ou selecione Padrão do sistema para usar as configurações globais de fuso horário.
Tema	Define a aparência do frontend: Padrão do sistema - use as configurações padrão do sistema Azul - tema azul padrão Escuro - tema escuro alternativo Luz de alto contraste - tema claro com alto contraste Escuro de alto contraste - tema escuro com alto contraste
Auto-login	Marque esta caixa de seleção para que o Zabbix se lembre do usuário e faça o login do usuário automaticamente por 30 dias. Os cookies do navegador são usados para isso.
Auto-logout	Com esta caixa de seleção marcada, o usuário será desconectado automaticamente, após a quantidade definida de segundos (mínimo de 90 segundos, máximo de 1 dia). Sufixos de tempo são suportados, por exemplo 90s, 5m, 2h, 1d. Observe que esta opção não funcionará: * Se a opção de configuração global "Mostrar aviso se o servidor Zabbix estiver inativo" estiver habilitada e o frontend do Zabbix for mantido aberto; * Quando as páginas do menu Monitoramento executam atualizações de informações de segundo plano; * Se efetuar login com a opção Lembrar-me por 30 dias marcada.
Atualizar	Defina a taxa de atualização usada para gráficos, dados de texto simples, etc. Pode ser definido como 0 para desabilitar.
Linhas por página	Você pode determinar quantas linhas por página serão exibidas nas listas.
URL (após o login)	Você pode fazer o Zabbix transferir o usuário para uma URL específica após o login bem-sucedido, por exemplo, para a página de Problemas.

Mídia do usuário

A guia Mídia contém uma lista de todas as mídias definidas para o usuário. A mídia é usada para enviar notificações. Clique em Adicionar para atribuir mídia ao usuário.

Consulte os **Tipos de mídia** seção para obter detalhes sobre como configurar a mídia do usuário.

Permissões

A guia Permissões contém informações sobre:

- A função do usuário. Os usuários não podem alterar sua própria função.
- O tipo de usuário (Usuário, Admin, Super Admin) definido na função configuração.
- Grupos de hosts aos quais o usuário tem acesso. Usuários do tipo 'Usuário' e 'Admin' não tem acesso a nenhum grupo de hosts e hosts por padrão. Para obter o acesso que eles precisam para serem incluídos em grupos de usuários que têm acesso aos respectivos grupos de hosts e hosts.
- Direitos de acesso a seções e elementos do frontend Zabbix, módulos, e métodos de API. Os elementos aos quais o acesso é permitido são exibidos na cor verde. A cor cinza claro significa que o acesso ao elemento é negado.
- Direitos para realizar certas ações. As ações permitidas são exibido na cor verde. A cor cinza claro significa que um usuário não tem o direito de realizar esta ação.

Consulte a página **Permissões do usuário** para obter detalhes.

2 Permissões

Visão geral

Você pode diferenciar as permissões do usuário no Zabbix definindo o respectiva função de usuário. Em seguida, os usuários sem privilégios precisam ser incluídos grupos de usuários que têm acesso aos dados do grupo de hosts.

Função do usuário

A função do usuário define quais partes da interface do usuário, quais métodos de API e quais ações estão disponíveis para o usuário. As seguintes funções são pré-definidas no Zabbix:

Tipo de usuário	Descrição
Função de convidado	O usuário tem acesso às seções do menu Monitoramento, Inventário e Relatórios, mas sem direitos para realizar nenhuma ação.
Função do usuário	O usuário tem acesso às seções do menu Monitoramento, Inventário e Relatórios. O usuário não tem acesso a nenhum recurso por padrão. Quaisquer permissões para grupos de hosts devem ser atribuídas explicitamente.
Função Admin	O usuário tem acesso às seções do menu Monitoramento, Inventário, Relatórios e Configuração. O usuário não tem acesso a nenhum grupo de hosts por padrão. Quaisquer permissões para grupos de host devem ser fornecidas explicitamente.
Função Super Admin	O usuário tem acesso a todas as seções do menu. O usuário tem acesso de leitura/gravação a todos os grupos de hosts. As permissões não podem ser revogadas negando acesso a grupos de hosts específicos.

Funções do usuário são configuradas na seção Administration→User roles. Super Admins podem modificar ou excluir funções predefinidas e criar mais funções com conjuntos de permissões personalizadas.

Para atribuir uma função ao usuário, vá para a guia Permissões no formulário de configuração do usuário, localize o campo Função e selecione uma função. Uma vez que uma função é selecionada uma lista de permissões associadas será exibida abaixo.

User
Media
Permissions

* Role

Admin role

Select

User type

Admin

Permissions

Host group
All groups

Permissions
None

Permissions can be assigned for user groups only.

Access to UI elements

Monitoring
Dashboard
Problems
Hosts
Overview
Latest data
Maps
Discovery
Services

Inventory
Overview
Hosts

Reports
Availability report
Triggers top 100
Notifications
Scheduled reports

Configuration
Host groups
Templates
Hosts
Maintenance
Actions
Discovery
Services

Access to modules
No enabled modules found.

Access to API
Enabled

Access to actions
Create and edit dashboards
Create and edit maps
Create and edit maintenance
Add problem comments
Change severity
Acknowledge problems
Close problems
Execute scripts
Manage API tokens
Manage scheduled reports

Add
Cancel

Permissões para hospedar grupos

O acesso a qualquer dado de host no Zabbix é concedido a **user groups** no grupo de hosts nível apenas.

Isso significa que um usuário individual não pode ter acesso direto a um host (ou grupo de hosts). Ele só pode ter acesso a um host sendo parte de um grupo de usuários que recebe acesso ao grupo de hosts que contém o hospedeiro.

3 Grupos de usuários

Visão geral

Os grupos de usuários permitem agrupar usuários para fins organizacionais e para atribuir permissões aos dados. Permissões para monitorar dados de grupos de hosts são atribuídos a grupos de usuários, não a usuários individuais.

Muitas vezes pode fazer sentido separar quais informações estão disponíveis para um grupo de usuários e o que - para outro. Isso pode ser realizado agrupando usuários e, em seguida, atribuindo permissões variadas a grupos de hosts.

Um usuário pode pertencer a qualquer número de grupos.

Configuração

Para configurar um grupo de usuários:

- Vá para Administração → Grupos de usuários
- Clique em Criar grupo de usuários (ou no nome do grupo para editar um grupo existente)
- Editar atributos de grupo no formulário

A guia **Grupo de usuários** contém atributos gerais de grupo:

The screenshot shows the 'User group' configuration form. The 'Group name' field is required and contains 'Security specialists'. The 'Users' field has a search bar with two users selected: 'Admin (Zabbix Administrator)' and 'user (New User)'. The 'Frontend access' dropdown is set to 'System default'. The 'Enabled' checkbox is checked, and the 'Debug mode' checkbox is unchecked. The 'Add' and 'Cancel' buttons are at the bottom.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome do grupo Usuários	Nome do grupo exclusivo. Para adicionar usuários ao grupo, comece digitando o nome de um usuário existente. Quando a lista suspensa com nomes de usuário correspondentes aparecer, role para baixo para selecionar. Como alternativa, você pode clicar no botão Selecionar para selecionar usuários em um pop-up.
Acesso de frontend	Como os usuários do grupo são autenticados. Padrão do sistema - use o método de autenticação padrão (definido globalmente) Interno - use a autenticação interna do Zabbix (mesmo que a autenticação LDAP seja usada globalmente). Ignorado se a autenticação HTTP for o padrão global. LDAP - use a autenticação LDAP (mesmo que a autenticação interna seja usada globalmente). Ignorado se a autenticação HTTP for o padrão global. Desativado - acesso ao frontend Zabbix é proibido para este grupo
Ativado	Status do grupo de usuários e membros do grupo. Marcado - grupo de usuários e usuários estão habilitados Desmarcado - grupo de usuários e usuários estão desabilitados
Modo de depuração	Marque esta caixa de seleção para ativar o modo de depuração para os usuários.

A guia **Permissões** permite especificar o acesso do grupo de usuários ao host dados do grupo (e, portanto, do host):

User group
Permissions
Tag filter

Permissions

Host group
All groups
Discovered hosts
Hypervisors
Linux servers
Templates (including subgroups)
Templates/Server hardware
Templates/Virtualization

Permissions
None

Read-write	Read	Deny	None
Read-write	Read	Deny	None
Read-write	Read	Deny	None
Read-write	Read	Deny	None
Read-write	Read	Deny	None
Read-write	Read	Deny	None

Read-write	Read	Deny	None
------------	------	------	------

☐ Include subgroups
[Add](#)

As permissões atuais para grupos de hosts são exibidas no bloco Permissões.

Se as permissões atuais do grupo de hosts forem herdadas por todos os grupos de hosts, isto é indicado pelo texto incluindo subgrupos no parênteses após o nome do grupo de hosts. Note que um usuário Super admin pode impor que grupos de hosts aninhados tenham o mesmo nível de permissões que o grupo de host pai; isso pode ser feito no fomrulário de **configuração** grupo de hosts. Você pode alterar o nível de acesso a um grupo de hosts:

- **Leitura-gravação** - acesso de leitura e gravação a um grupo de hosts;
- **Leitura** - acesso somente leitura a um grupo de hosts;
- **Negar** - acesso negado a um grupo de hosts;
- **Nenhum** - nenhuma permissão é definida.

Use o campo de seleção abaixo para selecionar grupos de hosts e o nível de acesso a eles. (observe que selecionar Nenhum removerá o grupo de hosts do a lista se o grupo já estiver na lista). Se você deseja incluir grupos de hosts aninhados, marque a caixa de seleção Incluir subgrupos. Este campo é auto-completar, então começar a digitar o nome de um grupo de hosts oferecerá uma lista suspensa de grupos correspondentes. Se você deseja ver todos os grupos de hosts, clique em em Selecionar.

Observe que é possível para usuários Super Admin no grupo de hosts **configuração** para impor o mesmo nível de permissões aos grupos de hosts aninhados que o grupo de hosts pai.

A guia **Filtro de tags** permite definir permissões baseadas em tags para usuários groups para ver os problemas filtrados pelo nome da tag e seu valor:

User group
Permissions
Tag filter

Permissions

Host group
Templates/Databases

Tags
Service: MySQL

Action
[Remove](#)

☐ Include subgroups
[Add](#)

Para selecionar um grupo de hosts ao qual aplicar um filtro de tags, clique em Selecionar para obter a lista completa de grupos de hosts existentes ou comece a digitar o nome de um host group para obter uma lista suspensa de grupos correspondentes. Se você deseja aplicar filtros de tags para grupos de hosts aninhados, marque Incluir subgrupos caixa de seleção.

O filtro de tags permite separar o acesso ao grupo de hosts do possibilidade de ver problemas.

Por exemplo, se um administrador de banco de dados precisar ver apenas "MySQL" problemas de banco de dados, é necessário criar um grupo de usuários para banco de dados administradores primeiro, depois especifique o nome da tag "Serviço" e o valor "MySQL".

Templates/Databases X
type here to search

Select

Service

MySQL

Se o nome da tag "Serviço" for especificado e o campo de valor for deixado em branco, o grupo de usuários correspondente verá todos os problemas do grupo de hosts selecionado com o nome da tag "Serviço". Se os campos de nome e valor da tag forem deixados em branco, mas o grupo de hosts selecionado, o grupo de usuários correspondente verá todos problemas para o grupo de hosts selecionado. Certifique-se de que um nome de tag e um valor de tag estejam especificado corretamente, caso contrário, um grupo de usuários correspondente não verá Quaisquer problemas.

Vamos rever um exemplo quando um usuário é membro de vários grupos de usuários selecionado. A filtragem neste caso usará a condição OR para tags.

Grupo de usuários A			Grupo de usuários B			Resultado visível para um usuário (membro) de ambos os grupos
Filtro de tags						
Grupo de hosts	Nome da tag	Valor da tag	Grupo de hosts	Nome da tag	Valor da tag	
Modelos/Bancos de Dados	Serviço	MySQL	Modelos/Bancos de Dados	Serviço	Oracle	
Modelos/Bancos de dados	em branco	em branco	Modelos/Bancos de dados	Serviço	Oracle	
não selecionado	em branco	em branco	Modelos/Bancos de dados	Serviço	Oracle	Serviço: Problemas com MySQL ou Oracle visíveis Todos os problemas visíveis <Serviço:Oracle> problemas visíveis

::: não importante Adicionando um filtro (por exemplo, todas as tags em um determinado grupo de hosts "Modelos/Bancos de Dados") resulta na impossibilidade de veja os problemas de outros grupos de hosts. :::

Acesso ao host de vários grupos de usuários

Um usuário pode pertencer a qualquer número de grupos de usuários. Esses grupos podem ter diferentes permissões de acesso aos hosts.

Portanto, é importante saber quais hosts um usuário sem privilégios poder acessar como resultado. Por exemplo, vamos considerar como o acesso para hospedar **X** (no Hostgroup 1) será afetado em várias situações para um usuário que está nos grupos de usuários A e B.

- Se o Grupo A tiver apenas acesso de Leitura ao Grupo de hosts 1, mas ao Grupo B Acesso de Leitura-gravação ao Hostgroup 1, o usuário receberá **Leitura-gravação** acesso a 'X'.

::: não importante As permissões de "leitura-gravação" têm precedência sobre Permissões de "leitura" a partir do Zabbix 2.2. :::

- No mesmo cenário acima, se 'X' estiver simultaneamente também em Hostgroup 2 que é **negado** ao Grupo A ou B, o acesso a 'X' será ser **indisponível**, apesar de um acesso de Leitura-gravação ao Hostgroup 1.
- Se o Grupo A não tiver permissões definidas e o Grupo B tiver um Read-write acesso ao Hostgroup 1, o usuário terá acesso **Read-write** 'X'.
- Se o Grupo A tiver Negar acesso ao Hostgroup 1 e o Grupo B tiver um Ler-escrever acesso ao Hostgroup 1, o usuário terá acesso a 'X' **negado**.

Outros detalhes

- Um usuário de nível Admin com acesso Read-write a um host não será capaz de vincular/desvincular modelos, se ele não tiver acesso ao Modelos grupo. Com acesso de Leitura ao grupo Modelos, ele poderá vincular/desvincular modelos ao host, no entanto, não verá qualquer modelo na lista de modelos e não poderá operar com modelos em outros lugares.

- Um usuário de nível Admin com acesso Read a um host não verá o host na lista de hosts da seção de configuração; no entanto, o anfitrião os gatilhos estarão acessíveis na configuração do serviço de TI.
- Qualquer usuário não superadministrador (incluindo 'convidado') pode ver mapas de rede como desde que o mapa esteja vazio ou tenha apenas imagens. Quando hosts, grupos de hosts ou gatilhos são adicionados ao mapa, as permissões são respeitadas.
- O servidor Zabbix não enviará notificações para usuários definidos como ação destinatários da operação se o acesso ao host em questão for explicitamente "negado".

13 Armazenamento de segredos

Visão geral

É possível armazenar algumas informações confidenciais secretamente na HashiCorp Mecanismo de segredos do Vault KV - Versão 2. Os segredos podem ser salvos para:

- valores de macro do usuário
- credenciais de acesso ao banco de dados

O Zabbix fornece acesso somente leitura aos segredos no Vault, assumindo que segredos são gerenciados por outra pessoa.

Valores de macro do usuário

É possível armazenar valores de macro do usuário secretamente no Vault.

Um "**segredo do Vault**" valor de uma macro de usuário contém um caminho de referência (como 'path:key', para exemplo "secret/zabbix:password").

Os seguintes comandos podem ser usados para definir o valor para o caminho mencionado no exemplo:

```
# Habilite o ponto de montagem "secret/" se ainda não estiver habilitado, observe que "kv-v2" deve ser usado
$ segredos do cofre habilitam -path=secret/ kv-v2

# Coloque o novo segredo com a senha da chave no ponto de montagem "secret/" e caminho "secret/zabbix"
$ vault kv colocar secret/zabbix password=<senha>

# Testa se o segredo foi adicionado com sucesso
$ vault kv obter segredo/zabbix

# Finalmente teste com Curl, observe que "data" precisa ser adicionado manualmente após o ponto de montagem
$ curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix
```

O valor secreto é recuperado pelo servidor Zabbix em cada atualização de dados de configuração e é armazenado no cache de configuração. O token de autenticação para um acesso somente leitura aos caminhos de referência deve ser fornecido na configuração do servidor (parâmetro 'VaultToken'). Se o o valor da macro não pode ser recuperado com sucesso o item correspondente usar o valor ficará incompatível.

Também é possível acionar a atualização de valores secretos do Vault, usando uma linha de comando 'secrets_reload' **opção**.

O proxy Zabbix nunca se comunica com o Vault para obter outros segredos além credenciais do banco de dados. Valores secretos no proxy Zabbix são recuperados de Servidor Zabbix em cada sincronização de configuração e armazenado na configuração cache da mesma forma que no servidor Zabbix.

Isso significa que um proxy Zabbix não pode iniciar a coleta de dados após uma reinicialização até receber a atualização dos dados de configuração do servidor Zabbix para a primeira vez. A criptografia deve ser habilitada entre o servidor Zabbix e procuração; caso contrário, uma mensagem de aviso do servidor é registrada.

Credenciais do banco de dados

É suportado para armazenar credenciais de banco de dados usadas pelo servidor Zabbix, proxies e frontend secretamente no Vault:

- Parâmetros relacionados ao cofre para recuperar credenciais de banco de dados podem ser opcionalmente inserido no frontend **instalação assistente**.

As credenciais do banco de dados recuperadas do Vault serão armazenadas em cache pelo a parte dianteira. Observe que o diretório de arquivos temporários do sistema de arquivos é usado para cache de credencial de banco de dados no frontend. Você pode usar o ZBX_DATA_CACHE_TTL **constante** para controlar com que frequência o cache de dados é atualizado/invalidado.

- Para servidor/proxy pode ser usado o parâmetro de configuração VaultDBPath para especificar o caminho de onde as credenciais para o banco de dados serão recuperado pelas chaves 'password' e 'username' (por exemplo: segredo/zabbix/banco de dados).

Os seguintes comandos podem ser usados para definir os valores para o caminho mencionado no exemplo:

```
# Habilite o ponto de montagem "secret/" se ainda não estiver habilitado, observe que "kv-v2" deve ser usado
$ segredos do cofre habilitam -path=secret/ kv-v2

# Coloque novos segredos com as chaves nome de usuário e senha no ponto de montagem "secret/" e caminho "s"
$ vault kv colocar secret/zabbix/database username=zabbix password=<senha>

# Testa se o segredo foi adicionado com sucesso
$ vault kv obter secret/zabbix/database

# Finalmente teste com Curl, observe que "data" precisa ser adicionado manualmente após o ponto de montagem "s"
$ curl --header "X-Vault-Token: <VaultToken>" https://127.0.0.1:8200/v1/secret/data/zabbix/database
```

Parâmetros de configuração

Para o servidor/proxy Zabbix, novos parâmetros de configuração foram adicionados para Autenticação de cofre e recuperação de credenciais de banco de dados:

- VaultToken - Token de autenticação do Vault (veja Zabbix [servidor/proxy](#) arquivo de configuração para detalhes)
- VaultURL - URL do servidor de cofre HTTP[S]
- VaultDBPath - Caminho do cofre de onde as credenciais para o banco de dados serão recuperado pelas chaves 'password' e 'username' (por exemplo: segredo/zabbix/banco de dados)

O servidor Zabbix e o proxy Zabbix lêem a configuração relacionada ao Vault parâmetros de zabbix_server.conf e zabbix_proxy.conf na inicialização.

O servidor Zabbix e o proxy Zabbix lerão adicionalmente "VAULT_TOKEN" variável de ambiente uma vez durante a inicialização e desmarque-a para que não estar disponível através de scripts bifurcados; é um erro se ambos VaultToken e VAULT_TOKEN contêm valor.

Note:

Barra e dois pontos são símbolos reservados. Barra só pode ser usado para separar o ponto de montagem do caminho (por exemplo, secret/zabbix onde o ponto de montagem é "secreto" e "zabbix" é o caminho) e, no caso de Macros de cofre, dois pontos só podem ser usados para separar o caminho da chave. Isto é possível codificar URL "/" e ":" se houver necessidade de criar ponto com nome separado por barra (por exemplo foo/bar/zabbix onde o ponto de montagem é "foo/bar" e o caminho é "zabbix" como "foo%2Fbar/zabbix") e se o nome ou caminho do ponto de montagem precisar conter cólon.

Configurando o TLS

O certificado assinado por uma autoridade de certificação (CA) deve ser adicionado ao o armazenamento CA padrão. Alternativamente, um local de armazenamento CA personalizado pode ser especificado usando o parâmetro de configuração SSLCALocation; note que em neste caso, o diretório de certificados deve ser preparado usando o openssl utilitário c_rehash, por exemplo, configure SSLCALocation e copie "ca.pem" dentro desse diretório, execute o seguinte comando:

```
$ c_rehash .
```

14 Relatórios programados

Versão geral

Esta seção fornece informações sobre configurações para agendamento de relatórios

Pré-requisitos:

- O serviço web Zabbix deve ser instalado e configurado corretamente para habilitar a geração de relatórios agendados - consulte [Configurando reports](#) para obter instruções.
- Um usuário deve ter um [usuário função](#) do tipo Admin ou Super admin com as seguintes permissões:
 - * //Scheduled reports// in the //Access to UI elements// block (to view reports);
 - * //Manage scheduled reports// in the //Access to actions// block (to create/edit reports).

Parameter	Descrição
Owner	Usuário que cria um relatório. Usuários de nível Super admin têm permissão para alterar o proprietário. Para usuários de nível Admin, este campo é apenas para leitura.
Name	Nome do relatório; deve ser único.
Dashboard	Painel no qual o relatório se baseia; apenas um painel pode ser selecionado por vez. Para selecionar um painel, comece digitando o nome - uma lista de painéis correspondentes será exibida; role a tela para baixo para selecionar. Se preferir, você pode clicar em Selecionar próximo ao campo e selecionar um painel a partir da lista exibida. Se o painel contiver várias páginas, somente a primeira será enviada como relatório.
Period	Período em que o relatório será preparado. Selecione o dia, semana, mês ou ano anterior.
Cycle	Frequência da geração do relatório. Os relatórios podem ser enviados diariamente, semanalmente, mensalmente, ou anualmente. O modo "Semanalmente" permite selecionar os dias da semana que o relatório será enviado.
Start time	Horário do dia em formato hh:mm quando o relatório será preparado.
Repeat on	Dias da semana quando o relatório será enviado. Este campo está disponível apenas se o Cycle estiver configurado para "Semanalmente".
Start date	Data em que geração regular do relatório deve ser iniciada.
End date	Data em que a geração regular do relatório deve ser parada.
Subject	Assunto do e-mail do relatório. Suporta {TIME} macro.
Message	Corpo do e-mail do relatório. Suporta {TIME} macro.
Subscriptions	Lista dos destinatários do relatório. Por padrão, inclui apenas o proprietário do relatório. Qualquer usuário do Zabbix com mídia de e-mail configurada pode ser especificado como destinatário do relatório. Clique em Adicionar usuário ou em Adicionar grupo de usuários para adicionar mais destinatários. Clique no nome do usuário para alterar as configurações: Gerar relatório por - se os dados do relatório devem ser gerados com base nas permissões do painel do usuário atual ou do destinatário. Status - selecione "Incluir" para enviar o relatório para o usuário ou "Excluir" para evitar de enviar o relatório para esse usuário. Pelo menos um usuário deve conter o status como "Incluir". O status "Excluir" pode ser utilizado para excluir usuários específicos de um grupo de usuários. Note que usuários com permissões insuficientes (ou seja, usuários com uma função baseada em usuário Admin que não são membros do mesmo grupo de usuários que o destinatário ou proprietário do relatório) verão "Usuário inacessível" ou "Grupo de usuários inacessível", em vez dos nomes reais nos campos Destinatário e Gerar relatório por; os campos Status e Action serão exibidos como somente leitura.
Enabled	Reportar status. Limpar essa caixa de seleção irá desabilitar o relatório.
Description	Um descrição opcional do relatório. Essa descrição é para uso interno e não será enviada para os destinatários do relatório.

Botões do formulário

Os botões na parte inferior do formulário permitem realizar diversas operações.

Add	Adicione um relatório. Este botão estará disponível apenas para relatórios novos.
Update	Atualize as propriedades de um relatório.
Clone	Crie outro relatório com base nas propriedades do relatório atual.
Test	Teste se a configuração do relatório está correta enviando um relatório para o usuário atual.
Delete	Exclua o relatório.
Cancel	Cancele a edição das propriedades do relatório.

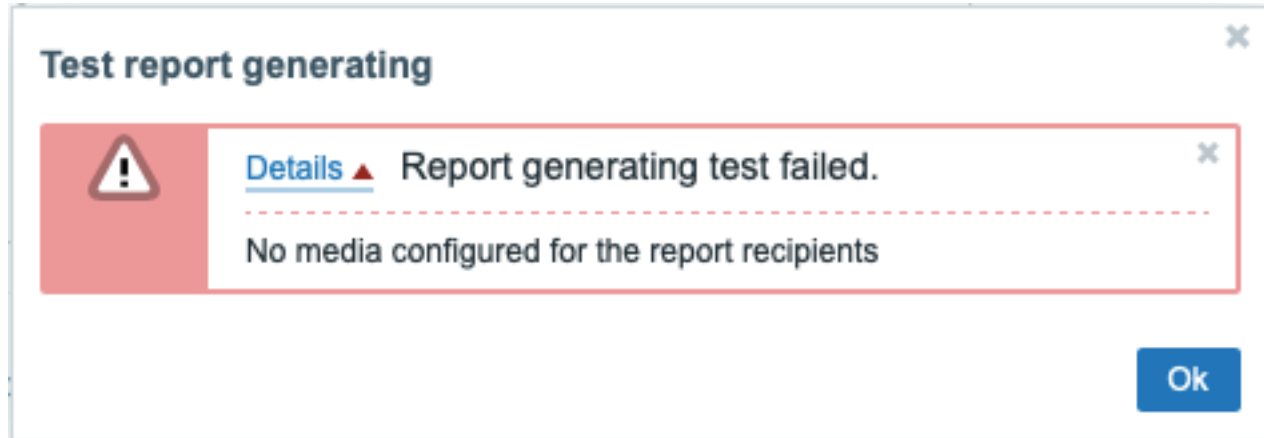
Testes

Para testar um relatório, clique no botão Test na parte inferior do formulário de configuração do relatório.

Note:

O botão Test não estará disponível se o formulário de configuração do relatório for aberto do menu **action menu**.

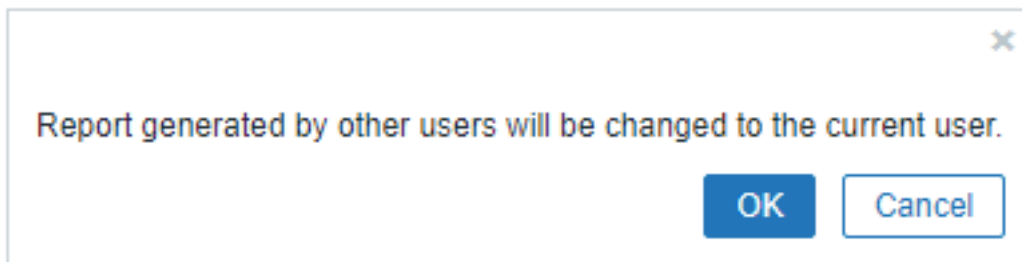
Se a configuração estiver correta, o relatório do teste é enviado imediatamente para o usuário atual. Para os relatórios de teste, assinantes e configurações de usuário para Generate report by são ignorados. Se a configuração estiver incorreta, uma mensagem de erro é exibida descrevendo a possível causa.



Atualizando um relatório

Para atualizar um relatório existente, clique no nome do relatório, faça as alterações necessárias na configuração, e em seguida clique no botão Atualizar.

Se um relatório existente for atualizado por outro usuário e esse usuário alterar o Dashboard, ao clicar no botão Atualizar, uma mensagem de alerta "Relatórios gerados por outros usuários serão alterados para o usuário atual" será exibida.



Clicando em OK, você verá as seguintes alterações:

- Configurações Gerar relatório por serão atualizadas para exibir o usuário que editou o relatório por último (a menos que Gerar relatório por estiver configurado para destinatário).
- Usuários que foram exibidos como "Usuário inacessível" ou "Grupo de usuários inacessível" serão excluídos da lista de assinantes do relatório.

Clicando em Cancelar, você fechará o formulário de configuração e cancelará a atualização do relatório.

Clonando um relatório

Para clonar um relatório rapidamente, clique no botão Clonar na parte inferior de um formulário de configuração de relatório existente. Ao clonar um relatório criado por outro usuário, o usuário atual torna-se o proprietário do novo relatório.

As configurações do relatório serão copiadas para o novo formulário de configuração do relatório, levando em consideração as permissões do usuário:

- Se o usuário que clona um relatório não tiver permissões para um painel, o campo Dashboard será limpo.
- Se o usuário que clona um relatório não tiver permissões para alguns usuários ou grupo de usuários na lista de Assinaturas, os destinatários inacessíveis não serão clonados.
- As configurações de Generate report by serão atualizadas para exibir o usuário atual (a menos que o Generate report by esteja definido para o destinatário).

Altere as configurações necessárias e o nome do relatório, e, em seguida, clique em Adicionar.

8. Monitoramento de serviço

Visão geral A funcionalidade de monitoramento de serviço é destinada àqueles que querem obter uma visão de alto-nível (negócio) da infraestrutura monitorada. Em muitos casos, nós não estamos interessados em detalhes de baixo-nível, como a falta de espaço em disco, alta carga de processador, etc. O que estamos interessados é na disponibilidade geral de um serviço fornecido pelo nosso departamento de TI. Podemos também estar interessados em identificar locais frágeis da nossa infraestrutura de TI, SLA de vários serviços de TI, a estrutura da infraestrutura de TI existente, e outras informações em um nível mais alto.

O monitoramento de serviço do Zabbix fornece respostas a todas as questões mencionadas.

O monitoramento de serviço permite criar uma representação hierárquica dos dados monitorados.

Uma estrutura de serviço muito simples pode parecer como:

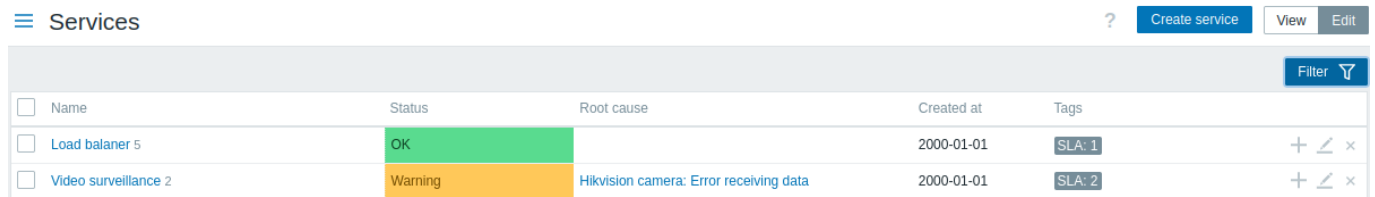
Serviço

```
|
|-Estações de trabalho
| |
| |-Estação de trabalho 1
| |
| |-Estação de trabalho2
|
|-Servidores
```

Cada nós da estrutura possui estados de atributo. O estado é calculado e propagado para níveis mais altos de acordo ao algoritmo selecionado. O estado de nós individuais é afetado pelo estado de problemas mapeados. O mapeamento de problema é realizado com etiquetagem (tagging).

1 Árvore de serviços

A Árvore de serviços é configurada Services->Services na seção menu. No canto superior direito, alterne do **View** para o modo de Edição.



Services					Filter
<input type="checkbox"/> Name	Status	Root cause	Created at	Tags	
<input type="checkbox"/> Load balancer 5	OK		2000-01-01	SLA: 1	+ ↗ ×
<input type="checkbox"/> Video surveillance 2	Warning	Hikvision camera: Error receiving data	2000-01-01	SLA: 2	+ ↗ ×

Para um novo serviço **configure**, clique no botão Create service que está no canto superior direito. Para adicionar um serviço child rapidamente, você também pode clicar em um ícone de adição próximo ao serviço Parent. Isso abrirá o mesmo formulário de configuração de serviço, mas o parâmetro serviço Parent estará pré-preenchido.

Configuração de serviço Na aba **Service**, especificar os parâmetros de serviço necessários:

Service

Service
Tags 2
Child services

* Name

Connections

Parent services

Availability
X

type here to search

Select

Problem tags

Name	Operation	Value	Action
Type	Equals	Connection	Remove

Add

* Sort order (0->999)

0

Status calculation rule

Most critical of child services

Description

Created at

2000-01-01

☐ Advanced configuration

Update
Clone
Delete
Cancel

Todos os campos de input obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Name	Nome Service.
Parent services	Os services Parent aos quais o serviço pertence. Deixe esse campo vazio caso você esteja adicionando o serviço de nível mais alto. Um serviço pode ter múltiplos serviços parent. Nesse caso, ele será exibido na árvore service abaixo de cada parent services.
Problem tags	Especifique as tags para mapear os dados do problema para service: Equals - inclua os nomes e valores de especificados (case-sensitive) Contains - inclua os nomes de tags especificados onde os valores de tag contêm a sequência inserida (substring match, case-insensitive) O nome da Tag correspondente é sempre case-sensitive.
Sort order	Ordem de classificação para exibição, o mais baixo primeiro.
Status calculation rule	Regra para cálculo do status de service: Mais crítico se todos os children tiverem problemas - o problema mais crítico no child service é utilizado para colorir o service status, se todos os children tiverem problemas. O mais crítico dos child services - o problema mais crítico nos child services é usado para colorir o service status Set status to OK - não calcular o service status
Description	Descrição Service.
Advanced configuration	Assinale a caixa de seleção abaixo Advanced configuration para configurar regras adicionais de cálculo de status.

Configuração avançada

☒ Advanced configuration

Additional rules

Name	Action
Average - If at least 4 child services have Average status or above	Edit Remove
Disaster - If at least 3 child services have High status or above	Edit Remove
Add	

Status propagation rule

Weight

Parâmetro	Descrição
Additional rules	Clique em Adicionar para definir as regras adicionais de cálculo.
Set status to	Defina service status para OK (padrão), Não classificado, Informação, Aviso, Médio, Alto ou Desastre* em caso de correspondência de condição.
Condition	Selecione a condição do child service direto: se pelo menos (N) child services tiverem (Status) ou acima se pelo menos (N%) dos child services tiverem (Status) ou acima se menos que (N) child services tiverem (Status) ou abaixo se menos de (N%) child services tiverem (Status) ou abaixo se o peso de child services com (Status) ou acima for pelo menos (W) se o peso de child services com (Status) ou acima for pelo menos (N%) se o peso de child services com (Status) ou abaixo for menos que (W) se o peso de child services com (Status) ou abaixo for menor que (N%)
N (W)	Se várias condições forem especificadas e a situação corresponder a mais de uma condição, a gravidade mais alta será definida.
Status	Definir o valor de N ou W (1-100000), ou N% (1-100) na condição.
Status propagation rule	Selecione o valor de Status na condição: OK (padrão), Não classificado, Informação, Aviso, Médio, Alto ou Desastre.
Status propagation rule	Regra para propagar o service status para o parent service: Como está - o status é propagado sem mudança Como aumentar - você pode aumentar o status propagado em 1 a 5 gravidades Como diminuir - você pode diminuir o status propagado em 1 a 5 gravidades Ignorar este service - o status não é propagado para o parents service de forma alguma Status fixo - o status é propagado estaticamente statically, i.e. sempre com o mesmo
Peso	Peso do service (número inteiro no intervalo de 0 (padrão) a 1000000).

Note:

As regras adicionais de cálculo de status podem ser usadas somente para aumentar o nível de gravidade sobre o nível calculado de acordo com o parâmetro principal de Status calculation rule. Se, de acordo com as regras de adicionais, o status deveria ser Warning (Aviso), de acordo com o status Disaster Status calculation rule, o service terá o status Disaster.

As abas **Tags** contém **service-level tags**. As tags de nível service são utilizadas para identificar um serviço. Tags desse tipo não são usadas para mapear problemas para o serviço. Para isso, use **Problem tags** na primeira aba).

A aba **Child services** permite especificar os serviços dependentes. Clique em Add para adicionar um serviço a partir da lista de serviços existentes. Se você deseja adicionar um novo child service, salve este serviço primeiro, e em seguida, clique no ícone de adição ao lado mais do serviço que você acabou de criar.

Tags Há dois tipos diferentes de tags em serviços:

- Tags serviço
- Tags problema

Tags serviço

As tags de serviço são utilizadas para corresponder serviços com **service actions** e **SLAs**. Essas tags são especificadas na guia de configuração Tags de serviços. Para mapear SLAs, é usada a lógica OU: um serviço será mapeado para um SLA se ele tiver pelo

menos uma tag correspondente. Em ações de serviço, as regras de mapeamento são configuráveis e podem usar lógica E, OU, ou E/OU.

Service Tags 1 Child services

Tags	Name	Value
	<input type="text" value="internal"/>	<input type="text" value="monitoring"/>
	<input type="text" value="tag"/>	<input type="text" value="value"/>
	Add	

Tags problema

As tags problema são usadas para associar problemas e serviços. Essas tags são especificadas na guia de configuração principal do serviço. Somente os serviços de child do nível hierárquico mais baixo podem ter tags de problema definidas e estar diretamente correlacionados aos problemas. Se as tags de problema corresponderem, o status do serviço será alterado para o mesmo status do problema. Em caso de vários problemas, o serviço terá o status de mais severo de todos. O status de um serviço parent é então calculado baseado nos status dos serviços de child de acordo com as regras de cálculo status.

Se múltiplas tags são especificadas, several tags, a lógica e é utilizada: um problema deve ter todas as tags especificadas na configuração do serviço para ser mapeado para o serviço.

Problem tags	Name	Operation	Value	Action
	<input type="text" value="Database"/>	<input type="text" value="Equals"/>	<input type="text" value="MySQL"/>	Remove
	<input type="text" value="Type"/>	<input type="text" value="Contains"/>	<input type="text" value="Server"/>	Remove
	Add			

Note:

A problem in Zabbix inherits tags from the whole chain of templates, hosts, items, web scenarios, and triggers. Any of these tags can be used for matching problems to services.

Um problema no Zabbix é que herda tags de toda cadeia de templates, hosts, itens, cenários web, e triggers. Qualquer uma dessas tags pode ser usada para fazer a correspondência de problemas com serviços.

Exemplo:

O problema Web camera 3 indisponível possui tagstype:video surveillance, floor:1st e name:webcam 3 e status Aviso.

O serviço **Web camera 3** tem apenas a tag problema especificada: name:webcam 3

Problem tags	Name	Operation	Value	Action
	<input type="text" value="name"/>	<input type="text" value="Equals"/>	<input type="text" value="webcam 3"/>	Remove
	Add			

O status do serviço mudará de OK para Aviso quando o problema for detectado.

Se o serviço **Web camera 3** tivesse tags de problema name:webcam 3 e floor:2nd, seu status não seria alterado quando o problema fosse detectado, porque as condições seriam apenas parcialmente atendidas.

Note:

Os botões descritos abaixo estarão visíveis somente quando a seção Services estiver no modo de

Modificar os serviços existentes

Para editar um serviço existente, pressione o ícone lápis ao lado do serviço.

Para clonar um serviço existente, clique no ícone lápis para abrir sua configuração e, em seguida, clique no botão Clone. Quando um serviço é clonado, os links de seu parent são preservados, enquanto os links child não.

Para excluir um serviço, clique no ícone x próximo a ele. Quando você exclui um serviço parent, seus serviços child não serão excluídos e serão movidos um nível acima na árvore de serviços. (Os children do primeiro nível receberão o mesmo nível do parent excluído).

Dois botões abaixo da lista de serviços oferecem algumas opções de edição em massa:

- Atualizar em massa - atualizar propriedades de serviço em massa
- Excluir - excluir serviços

Para usar essas opções, marque as caixas de seleção antes dos serviços correspondentes, e, em seguida, clique no botão necessário.

2 Ações de serviço

Visão geral Nesta seção você pode visualizar e configurar as ações de serviço **actions**.

As ações de serviço são úteis se você deseja que algumas operações ocorram como resultado da mudança no status do serviço (OK ↔ PROBLEM), por exemplo:

- enviar mensagem
- reiniciar webserver

As ações de serviço são funcionalmente semelhantes a outros tipos de ação no Zabbix (por exemplo, ações trigger).

Configuração Para criar uma ação nova de serviço, vá para Service actions na subseção do menu Services, em seguida clique em Create action no canto superior direito.

As ações de serviço são configuradas da mesma forma que outros tipos de ações no Zabbix. Para mais detalhes, veja configurações **actions**.

As principais diferenças são:

- O acesso do usuário às ações de um serviço depende dos direitos de acesso aos serviços concedidos pelo papel do usuário **role**.
- As ações do serviço suportam diferentes conjuntos de condições **conditions**.

Condições As seguintes condições podem ser utilizadas nas ações de serviço:

Tipo de condição	Operadores suportados	Descrição
Service	equals does not equal	Especifique um serviço ou um serviço para excluir. equals - evento pertence a esse serviço. does not equal - evento não pertence a esse serviço.
Service name	contains does not contain	Especificar um service parent implicitamente seleciona todos os service child. Para especificar somente o parent service, todos os serviços aninhados precisam ser configurados adicionalmente com o operador does not equal . Especifique uma string no service name ou uma string para excluir. contains - o evento é gerado por um serviço que contém uma string no nome. does not contain - essa string não pode ser encontrada no service name.
Service tag name	equals does not equal contains does not contain	Especifique uma tag evento ou uma tag evento para excluir. As tags de evento de serviço podem ser definidas na seção de configuração do serviço Tags. equals - evento possui essa tag does not equal - evento não possui essa tag contains - evento possui uma tag que contém essa string does not contain - evento não possui uma tag que contém essa string.

Tipo de condição	Operadores suportados	Descrição
Service tag value	equals does not equal contains does not contain	Especifique uma combinação de tag de evento e valor ou uma combinação de tag e valor a ser excluída. As tags de evento do serviço podem ser definidas na seção de configuração do serviço Tags. equals - evento possui essa tag e valor does not equal - evento não possui dessa tag e valor contains - evento possui uma tag e valor que contém essas strings does not contain - evento não possui uma tag e um valor que contém essas strings.

Attention:

Certifique-se de definir **message templates** para Ações de serviço no menu Administration->Media types. Do contrário, as notificações não serão enviadas.

3 SLA

Visão geral Uma vez que os serviços **services** são criados, você pode iniciar a monitorar se o desempenho deles está de acordo com o acordo de nível do serviço (SLA).

A seção do menu do Services->SLA permite configurar os SLAs para diversos serviços. Um SLA do Zabbix define o objetivo do nível de serviço (SLO), com o cronograma esperado de tempo de atividade e os períodos planejados de inatividade. O SLA e os serviços são correspondentes ao serviço de tags **service tags**. O mesmo SLA pode ser aplicado a vários serviços - o desempenho será medido separadamente para cada serviço correspondente. Um único serviço pode ter múltiplos SLA atribuídos, o dado para cada SLA será exibido separadamente. Nos relatórios SLA, o Zabbix fornece dados do Indicador de Nível de Serviço (SLI), que mede a real disponibilidade do serviço. Se um serviço atende as tags SLA, esse serviço é determinado comparando o SLO (disponibilidade esperada em %) com SLI (disponibilidade real em %).

Configuração Para criar um SLA novo, clique no botão Criar SLA.

A aba **SLA** permite especificar os parâmetros gerais de SLA.

New SLA

SLA

Excluded downtimes

* Name

SLA:1

* SLO

99.9

%

Reporting period

Daily

Weekly

Monthly

Quarterly

Annually

Time zone

System default: (UTC+00:00) UTC

Schedule

24x7

Custom

* Effective date

2000-01-01

* Service tags

Name	Operation	Value	Action
SLA	Equals	1	Remove

Add

Description

Enabled

☒

Add

Cancel

Parameter	Descrição
Name	Digite o nome do SLA.
SLO	Digite o objetivo do nível do serviço (SLO) como porcentagem.
Reporting period	Selecione o período que afetará quais períodos serão usados em SLA report - diário, semanal, mensal, trimestral, ou anual.
Time zone	Selecione o fuso horário do SLA.
Schedule	Selecione o cronograma do SLA - 24x7 ou personalizado.
Effective date	Selecione a data de início do cálculo de SLA.
Service tags	Adicione as tags de serviço para identificar os serviços aos quais este SLA deve ser aplicado. Nome - o nome da tag de serviço deve corresponder exatamente, sensível a maiúsculas e minúsculas. Operação - selecione Igual se o valor da tag tiver que corresponder exatamente (maiúscula-minúscula) ou selecione Conter se parte do valor da tag corresponder (maiúscula-minúscula). Valor - se o valor da tag de serviço para pesquisa estiver de acordo com a operação selecionada. O SLA é aplicado a um serviço, se ao menos uma tag de serviço corresponder.
Description	Adicione uma descrição para o SLA.
Enabled	Assinale a caixa de seleção para habilitar o cálculo do SLA.

A aba de **Excluded downtimes** permite especificar os períodos de inatividade que são excluídos no cálculo do SLA.

New SLA

SLA Excluded downtimes 1

Start time	Duration	Name	Action
2022-02-01 02:00	3h	Maintenance	Edit Remove

[Add](#)

Add **Cancel**

Clique em Adicionar para configurar os períodos de inatividades excluídos, em seguida digite o nome do período, data de início e duração.

Relatórios SLA Como um serviço desempenha comparado a um SLA está visível no relatório de SLA **SLA report**. Os relatórios SLA podem ser visualizados:

- a partir da seção SLA, clicando no hyperlink do relatório do SLA;
- a partir da seção Services, clicando no nome do SLA na aba de informação;
- no Dashboard **SLA report**.

Uma vez que um SLA for configurado, a aba de Info na seção de serviços também exibirá algumas informações sobre o desempenho do serviço.

4 Exemplo Setup

Visão geral Esta seção descreve uma configuração simples para monitorar um cluster de alta disponibilidade do Zabbix como um serviço.

Pré-requisitos Antes configurar o monitoramento de serviço, é necessário ter os hosts configurados:

- HA node 1 com pelo menos um trigger e uma tag (preferencialmente, configurar em um nível de trigger) `component:HA node 1`
- HA node 2 com pelo menos um trigger e uma tag (preferencialmente, configurar em um nível de trigger) `component:HA node 2`

Árvore de serviços O próximo passo é criar uma árvore de serviço. Neste exemplo, a infraestrutura é muito básica e consiste em três serviços: Zabbix cluster (pai) e dois serviços filho Zabbix server node 1 and Zabbix server node 2.

```
Zabbix cluster
|
|- Zabbix server node 1
|- Zabbix server node 2
```

Na página de Serviços, ative o modo Edit e aperte em Criar serviço:

Create service

View

Edit

Na janela de configuração do serviço, digite nome Zabbix cluster e marque a caixa Advanced configuration.

New service

?

×

Service

Tags

Child services

* Name

Zabbix cluster

Parent services

type here to search

Select

Problem tags

Name	Operation	Value	Action
tag	Equals	value	Remove

Add

* Sort order (0->999)

0

Status calculation rule ⓘ

Most critical of child services

Description

☒ Advanced configuration

Additional rules

Name	Action
Add	

Status propagation rule

As is

Weight

0

Add

Cancel

Configurar regra adicional:

New additional rule



Set status to Disaster

Condition If at least **N** child services have **Status** status or above

N 2

Status Warning

Add

Cancel

Zabbix cluster terá dois serviços filho - uma para cada HA nodes. Se ambos HA nodes tiverem problemas de status Warning, o status de serviço pai deve ser configurado para Disaster. Para ativar, a regra adicional deve ser configurada como :

- Configurar status para: Disaster
- Condição: se pelo menos N serviços filho tem Status ou acima
- N: 2
- Status: Warning

Mude para Tags tab e adicione a tag Zabbix:server. T Essa tag será usada mais tarde para ações de serviços e relatórios de SLA.

New service



Service **Tags 1** Child services

Tags

Name	Value	Action
Zabbix	server	Remove

Add

Add

Cancel

Salve o serviço novo.

Para adicionar um serviço filho, clique no ícone de adição perto do serviço cluster Zabbix (o ícone é visível somente no modo Editar).

<input type="checkbox"/> Name	Status	Root cause	Created at	Tags	
<input type="checkbox"/> Zabbix cluster	OK		2022-05-10	Zabbix:server	+ ↗ ×

Displaying 1 of 1 found

Na janela de configuração do serviço, digite o nome Zabbix server node 1. Note que o parâmetro de serviços Parent (pai) já está pré-preenchido com Zabbix cluster.

A disponibilidade desse serviço é afetada por problemas no host HA node 1, marcados como tag problema `component:HA node 1`.

No parâmetro de tags problema, digite:

- Nome: componente
- Operação: Equals
- Valor: HA node 1

New service

? X

Service Tags Child services

* Name Zabbix server node 1

Parent services

Zabbix cluster X
type here to search

Select

Problem tags

Name	Operation	Value	Action
component	Equals	HA node 1	Remove
Add			

* Sort order (0->999)

0

Status calculation rule

Most critical of child services

Description

☐ Advanced configuration

Add

Cancel

Mude para Tags e adicione um serviço de tag: Zabbix server:node 1. Essa tag será utilizada posteriormente para ações de serviço e relatórios de SLA.

New service

? X

Service Tags 1 Child services

Tags

Name	Value	Action
Zabbix server	node 1	Remove
Add		

Add

Cancel

Salve o serviço novo. Para acionar o serviço filho, clique no ícone de adição perto do serviço cluster Zabbix (o ícone é visível somente no modo Edit)

<input type="checkbox"/> Name	Status	Root cause	Created at	Tags	
<input type="checkbox"/> Zabbix cluster	OK		2022-05-10	Zabbix: server	+ ✎ X

Displaying 1 of 1 found

Na janela de configuração do serviço, digite o nome, Zabbix server node 1. Observe que o parâmetro de serviços Parent (pai) já está pré-preenchido com Zabbix cluster.

A disponibilidade desse serviço é afetada por problemas no host HA node 1, como tag problema component:HA node 1. No parâmetro de tags problema, digite:

- Name: component
- Operation: Equals
- Value: HA node 1

New service

? X

Service **Tags** Child services

* Name

Parent services X Select
type here to search

Name	Operation	Value	Action
<input type="text" value="component"/>	<input type="text" value="Equals"/> ▼	<input type="text" value="HA node 1"/>	Remove
Add			

* Sort order (0->999)

Status calculation rule ⓘ ▼

Description

☐ Advanced configuration

Add

Cancel

Mude para Tags e adicione um serviço de tag: Zabbix server:node 1. Essa tag será utilizada posteriormente para ações de serviço e relatórios de SLA.

New service

? X

Service **Tags 1** Child services

Name	Value	Action
<input type="text" value="Zabbix server"/>	<input type="text" value="node 1"/>	Remove
Add		

Add

Cancel

Salve o serviço novo.

Crie um outro Child service do cluster Zabbix com o nome "Zabbix server node 2".

Definir as tags problema como:

- Nome: componente
- Operação: Equals
- Valor: HA node 2

Mude para a guia Tags e adicione uma tag serviço: Zabbix server:node 2.

Salve o serviço novo.

SLA Neste exemplo, o desempenho esperado do cluster do Zabbix é de 100%.

Primeiro, você precisa adicionar um novo acordo de nível de serviço.

Vá para a seção do menu Services->SLA e clique em Criar SLA. Digite o nome Zabbix cluster performance e defina o SLO para 100%.

O serviço do cluster Zabbix possui uma tag serviço Zabbix:server. Para utilizar esta SLA para medir o desempenho do cluster do Zabbix, no parâmetro Service tags, especifique:

- Nome: Zabbix
- Operação: Equals
- Valor: server

New SLA

SLA Excluded downtimes

* Name

Zabbix cluster performance

* SLO

100

%

Reporting period

Daily

Weekly

Monthly

Quarterly

Annually

Time zone

System default: (UTC+00:00) UTC

Schedule

24x7

Custom

* Effective date

2022-05-10

* Service tags

Name	Operation	Value	Action
Zabbix	Equals	server	Remove

Add

Description

Add

Cancel

Em uma configuração do mundo real, você também pode atualizar o período de relatórios desejado, fuso horário e data de início ou alterar o agendamento de 24/7 para personalizado. Para esse exemplo, as configurações padrões são o suficiente.

Mude para a guia Excluded downtimes e adicione paradas para períodos de manutenção agendada para excluí-los dos cálculos do SLA. Na seção Excluded downtimes, clique no link Adicionar, insira o nome downtime, a hora de início planejada e a duração.

New SLA

SLA Excluded downtimes 2

Excluded downtimes

Start time	Duration	Name	Action
2022-01-03 08:00	1h	Maintenance Jan	Edit Remove
2022-07-06 16:00	1h	Maintenance Jul	Edit Remove

Add

Add

Cancel

Clique em Adicionar para salvar o novo SLA.

Mude para seção de relatórios de SLA para visualizar o relatório de SLA para o cluster do Zabbix.

Year	SLO	SLI	Uptime	Downtime	Error budget
2022	100%	100	36m 53s	0	0

As informações de SLA também podem ser verificadas na seção Services.

All services / Zabbix cluster

Zabbix cluster

Parent services:

Status: OK

SLA: Zabbix cluster performance: 100 ?

Tags: Zabbix: server

Name	Status	Rc
Zabbix server node 1	OK	
Zabbix server node 2	OK	

9. Monitoramento web

Visão geral Com o Zabbix você pode verificar vários aspectos de disponibilidade de sites web.

Attention:

Para executar o monitoramento web o Zabbix Server deve ser inicialmente **configurado** com suporte a cURL (libcurl).

Para ativar o monitoramento web você precisa definir cenários web. Um cenário web consiste de uma ou várias requisições HTTP ou "etapas". As etapas são executadas periodicamente pelo Zabbix Server em uma ordem pré-definida. Se um host é monitorado pelo Proxy, as etapas são executadas pelo Proxy.

Os cenários web são anexados aos hosts/templates da mesma forma que os itens, gatilhos, etc. Isto significa que os cenários web podem também ser criados no nível de modelo e então aplicados a múltiplos hosts em um movimento.

As seguintes informações são coletadas em qualquer cenário web:

- média de velocidade de download por segundo para todas as etapas para o cenário completo
- número da etapa que falhou
- última mensagem de erro

As seguintes informações são coletadas em qualquer etapa cenário web:

- velocidade de download por segundo
- tempo de resposta
- código de resposta

Para mais detalhes, veja **itens de monitoramento web**.

Os dados coletados pela execução dos cenários web são armazenados no banco de dados. Os dados são automaticamente usados para gráficos, gatilhos e notificações.

O Zabbix pode também verificar se uma página HTML recuperada contém uma string pré-definida. Ele pode executar uma autenticação simulada e seguir um caminho simulado de clicks de mouse na página.

O monitoramento web do Zabbix suporta ambos HTTP e HTTPS. Quando executando um cenário web, o Zabbix opcionalmente seguirá redirecionamentos (veja a opção Seguir redirecionamentos abaixo). O número máximo de redirecionamentos é gravado

em código como 10 (usando a opção cURL [CURLOPT_MAXREDIRS](#)). Todos os cookies são preservados durante a execução de um cenário único.

Veja também [problemas conhecidos](#) para monitoramento web usando protocolo HTTPS.

Configurando um cenário web Para configurar um cenário web:

- Vá para: Configuração → Hosts (ou Modelos)
- Clique em Web na linha do host/modelo
- Clique em Criar cenário à direita (ou no nome do cenário para editar um cenário existente)
- Informe os parâmetros do cenário no formulário

A aba **Cenário** permite que você configure os parâmetros gerais de um cenário web.

ScenarioStepsTagsAuthentication

*

Name

Availability of example.com

*

Update interval

1m

*

Attempts

1

Agent

Zabbix

HTTP proxy

[protocol://][user[:password]@]proxy.example.com[:port]

Variables

Name

Value

name

⇒

value

Remove

Add

Headers

Name

Value

name

⇒

value

Remove

Add

Enabled

☒

Add

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetros de cenário:

Parâmetro	Descrição
Host	Nome do host/modelo ao qual o cenário pertence.
Name	Nome único de cenário.
Intervalo de atualização	Com que frequência o cenário será executado. Sufixos de tempo são suportados, p.e. 30s, 1m, 2h, 1d. Macros de usuário são suportadas. Note que se uma macro de usuário é usada e seu valor é alterado (p.e. 5m → 30s), a próxima verificação será executada de acordo com o valor anterior (distante no futuro com os valores de exemplo).

Parâmetro	Descrição
Tentativas	<p>O número de tentativas para execução das etapas do cenário web. No caso de problemas de rede (tempo esgotado, sem conectividade, etc.) o Zabbix pode repetir a execução de uma etapa por várias vezes. O conjunto de figuras afetará igualmente cada etapa do cenário. Até 10 tentativas podem ser especificadas, o valor padrão é 1.</p> <p>Nota: o Zabbix não repetirá uma etapa por causa de um código de resposta errado ou incompatibilidade de uma string requerida.</p>
Agente	<p>Selecione um agente cliente.</p> <p>O Zabbix simulará ser o navegador selecionado. Isto é útil quando um site web retorna conteúdo diferente para diferentes navegadores.</p>
Proxy HTTP	<p>Macros de usuário podem ser usadas neste campo.</p> <p>Você pode especificar um proxy HTTP para usar, utilizando o formato [protocolo://] [usuario[:senha]@]proxy.exemplo.com[:porta]. Isto define a opção cURL CURLOPT_PROXY.</p> <p>O prefixo opcional protocolo:// pode ser usado para especificar protocolos de proxy alternativos (o suporte a prefixo de protocolo foi adicionado no cURL 7.21.7). Sem protocolo especificado, o proxy será tratado como um proxy HTTP.</p> <p>Por padrão, a porta 1080 será usada.</p> <p>Se especificado, o proxy irá sobrescrever as variáveis de ambiente relacionadas como http_proxy, HTTPS_PROXY. Se não especificado, o proxy não irá sobrescrever variáveis de ambiente relacionadas ao proxy. O valor informado é passado assim "como é", sem verificação de coerência sendo executada.</p> <p>Você pode também informar uma endereço de proxy SOCKS. Se você especificar o protocolo errado, a conexão falhará e o item se tornará não suportado.</p> <p>Note que apenas autenticação simples é suportada com proxy HTTP.</p>
Variáveis	<p>Macros de usuário podem ser usadas neste campo.</p> <p>Variáveis que podem ser usadas em etapas de cenário (URL, variáveis de post).</p> <p>Elas têm o seguinte formato:</p> <p>{macro1}=valor1</p> <p>{macro2}=valor2</p> <p>{macro3}=regex:<expressão regular></p> <p>Por exemplo:</p> <p>{username}=Alexei</p> <p>{password}=kj3h5kj34bd</p> <p>{hostid}=regex:hostid é ([0-9]+)</p> <p>As macros podem então ser referenciadas nas etapas como {username}, {password} e {hostid}. O Zabbix automaticamente as substituirá com os valores atuais. Note que as variáveis com regex: precisam de uma etapa para obter o valor da expressão regular então o valor extraído só podem ser aplicados à etapa posterior.</p> <p>Se a parte do valor iniciar com regex: então a parte após ela é tratada como uma expressão regular que faz uma busca na página web e, se encontrada, armazena a correspondência na variável. Ao menos um subgrupo deve estar presente de modo que o valor correspondido possa ser extraído.</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p> <p>As variáveis são automaticamente codificadas para URL quando usadas nos campos de consulta ou dados de formulário para variáveis post, mas devem ser codificadas para URL manualmente quando usadas em postagem bruta ou diretamente na URL.</p>

Parâmetro	Descrição
Cabeçalhos	<p>Cabeçalhos HTTP customizados que serão enviados quando executando uma requisição.</p> <p>Cabeçalhos devem ser listados usando a mesma sintaxe em que eles apareceriam no protocolo HTTP, opcionalmente usando alguns atributos adicionais suportados pela opção cURL CURLOPT_HTTPHEADER.</p> <p>Por exemplo:</p> <p>Accept-Charset=utf-8</p> <p>Accept-Language=en-US</p> <p>Content-Type=application/xml; charset=utf-8</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p>
Habilitado	O cenário está ativo se esta caixa estiver selecionada, caso contrário - desabilitado.

Note que quando editando um cenário existente, dois botões extras estão disponíveis no formulário:

Clone	Cria outro cenário baseado nas propriedades do existente.
Clear history and trends	Elimina histórico e dados de tendência para o cenário. Isto fará com que o Server execute o cenário imediatamente após eliminar os dados.

Note:

Se o campo HTTP proxy for deixado vazio, outra forma para usar um proxy HTTP é definir variáveis de ambiente relacionadas ao proxy.

Para verificações HTTP - defina a variável de ambiente **http_proxy** para o usuário Zabbix Server. Por exemplo, `http_proxy=http://proxy_ip:proxy_port`.

Para verificações HTTPS - defina a variável de ambiente **HTTPS_PROXY**. Por exemplo, `HTTPS_PROXY=http://proxy_ip:proxy_port`. Mais detalhes estão disponíveis executando um comando shell: `# man curl`.

A aba **Etapas** permite que você configure as etapas do cenário web. Para adicionar uma etapa de cenário web, clique em Adicionar no bloco Etapas.

Scenario

Steps 2

Tags

Authentication

* Steps

Name	Timeout	URL	Required	Status
1: Site availability	15s	http://www.example.com		200
2: About	15s	http://www.example.com/about		200
Add				

Note:

Macros de usuário secretas não devem ser usadas nas URLs pois elas serão resolvidas para "*****".

×

Step of web scenario

*

Name

Site availability

*

URL

http://www.example.com

Parse

Query fields

Name

Value

name

⇒

value

Remove

Add

Post type

Form data

Raw data

Post fields

Name

Value

name

⇒

value

Remove

Add

Variables

Name

Value

name

⇒

value

Remove

Add

Headers

Name

Value

name

⇒

value

Remove

Add

Follow redirects

☒

Retrieve mode

Body

Headers

Body and headers

*

Timeout

15s

Required string

pattern

Required status codes

200

Update

Cancel

Configurando etapas

Parâmetros da etapa:

Parâmetro	Descrição
Nome	Nome de etapa único.

Parâmetro	Descrição
URL	<p>URL à qual conectar e recuperar dados. Por exemplo:</p> <p>https://www.example.com</p> <p>http://www.example.com/download</p> <p>Nomes de domínio podem ser especificados em caracteres Unicode. Eles são automaticamente convertidos com punycode para ASCII quando executando a etapa do cenário web.</p> <p>O botão Analisar pode ser usado para separar campos de consulta opcionais (como ?name=Admin&password=mypassword) da URL, movendo os atributos e valores para dentro dos Campos de consulta para codificação para URL automática.</p> <p>Variáveis podem ser usadas na URL, usando a sintaxe de {macro}.</p> <p>Variáveis podem ser codificadas para URL manualmente usando a sintaxe {{macro}}.urlencode().</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p> <p>Limitado a 2048 caracteres.</p>
Campos de consulta	<p>Variáveis HTTP GET para a URL.</p> <p>Especificado como pares de atributo e valor.</p> <p>Valores são automaticamente codificados para URL. Valores das variáveis do cenário, macros de usuário ou macros {HOST.*} são resolvidas e então codificadas para URL automaticamente. O uso de uma sintaxe {{macro}}.urlencode() fará uma dupla codificação.</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p>
Post	<p>Variáveis HTTP POST.</p> <p>No modo Form data, especificado como pares de atributo e valor. Os valores são codificados para URL automaticamente. Os valores das variáveis de cenário, macros de usuário ou macros {HOST.*} são resolvidas e então codificadas para URL automaticamente.</p> <p>No modo Raw data, atributos/valores são exibidos em uma linha única e concatenado com um símbolo &.</p> <p>Valores brutos podem ser codificados/decodificados para URL manualmente usando uma sintaxe {{macro}}.urlencode() ou {{macro}}.urldecode().</p> <p>Por exemplo: id=2345&userid={user}</p> <p>Se {user} é definido como uma variável do cenário web, ela será substituída pelo seu valor quando a etapa é executada. Se você deseja codificar a variável para URL, substitua {user} por {{user}}.urlencode().</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p>
Variáveis	<p>Variáveis em nível de etapa que podem ser usadas para funções GET e POST.</p> <p>Especificado como pares de atributo e valor.</p> <p>Variáveis em nível de etapa sobrescrevem variáveis em nível de cenário ou variáveis de etapas anteriores. No entanto, o valor de uma variável em nível de etapa afeta apenas a etapa posterior (e não a etapa atual).</p> <p>Elas tem o seguinte formato:</p> <p>{macro}=value</p> <p>{macro}=regex:<expressão regular></p> <p>Para mais informações veja a descrição da variável no nível do cenário.</p> <p>Variáveis são automaticamente codificadas para URL quando usadas em campos de consulta ou formulários de dados para variáveis de postagem, mas devem ser codificadas para URL manualmente quando usadas em postagem bruta ou diretamente na URL.</p>

Parâmetro	Descrição
Cabeçalhos	<p>Cabeçalhos HTTP customizados que serão enviados quando executando uma requisição.</p> <p>Especificado como pares de atributo ou valor.</p> <p>Cabeçalhos no nível da etapa irão sobrescrever os cabeçalhos especificados para o cenário.</p> <p>Por exemplo, definindo um atributo 'User-Agent' com valor ausente removerá o valor User-Agent definido no nível do cenário.</p> <p>Macros de usuário e macros {HOST.*} são suportados.</p> <p>Isto define a opção cURL CURLOPT_HTTPHEADER.</p>
Seguir redirecionamentos	<p>Marque a caixa de seleção para seguir redirecionamentos HTTP.</p> <p>Isto define a opção cURL CURLOPT_FOLLOWLOCATION.</p>
Modo de recuperação	<p>Selecione o modo de recuperação:</p> <p>Body - recupera apenas o corpo da resposta HTTP</p> <p>Headers - recupera apenas cabeçalhos da resposta HTTP</p> <p>Body and headers - recupera corpo e cabeçalhos da resposta HTTP</p>
Tempo limite	<p>O Zabbix não gastará mais do que a quantidade de tempo definida no processamento da URL (de um segundo ao máximo 1 hora).</p> <p>Atualmente este parâmetro define o tempo máximo para fazer a conexão à URL e o tempo máximo para executar uma requisição HTTP. Portanto, o Zabbix não gastará mais do que 2 x Tempo limite segundos na etapa.</p> <p>Sufixos de tempo são suportados, p.e. 30s, 1m, 1h. Macros de usuário são suportadas.</p>
String requerida	<p>Padrão de expressão regular requerida.</p> <p>A menos que o conteúdo recuperado (HTML) corresponda ao padrão requerido a etapa falhará. Se vazio, nenhuma verificação na string requerida é executada.</p> <p>Por exemplo:</p> <p>Homepage of Zabbix</p> <p>Welcome.*admin</p> <p>Nota: Referência à expressões regulares criadas no Zabbix Frontend não é suportada neste campo.</p> <p>Macros de usuário e macros {HOST.*} são suportadas.</p>
Códigos de estado requeridos	<p>Lista de códigos de estado HTTP. Se o Zabbix obtém um código que não está na lista, a etapa falhará.</p> <p>Se vazio, nenhuma verificação nos códigos de estado é executada.</p> <p>Por exemplo: 200,201,210-299</p> <p>Macros de usuário são suportadas.</p>

Note:

Quaisquer mudanças nas etapas do cenário web só serão salvas quando o cenário por completo for salvo.

Veja também um **exemplo da vida real** de como as etapas de monitoramento web podem ser configuradas.

Configurando etiquetas A aba de **Etiquetas** permite definir **etiquetas** no nível de cenário.

Scenario
Steps 2
Tags 1
Authentication

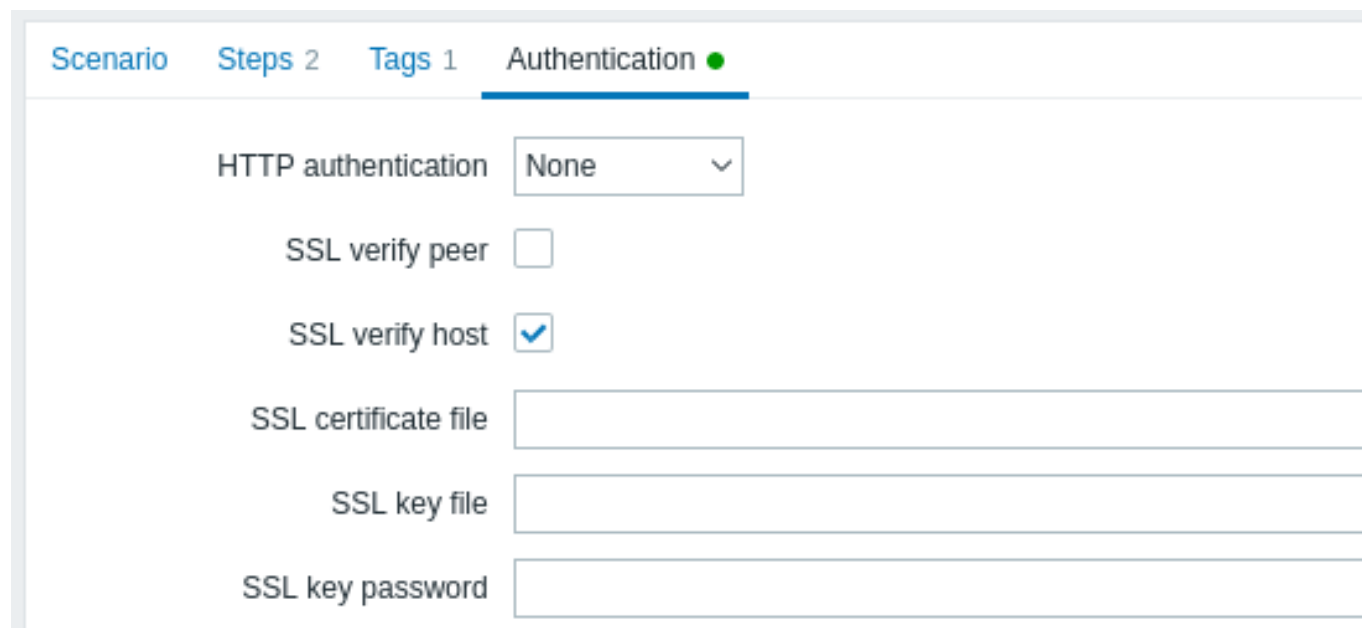
Scenario tags
Inherited and scenario tags

Name	Value	Action
Application	Web checks	Remove

Add

A etiquetagem permite filtrar cenários web e **itens** de monitoramento web.

Configurando autenticação A aba de **Autenticação** permite que você configure as opções de autenticação de cenário. Um ponto verde próximo ao nome da aba indica que algum tipo de autenticação HTTP está habilitada.



Parâmetros de autenticação:

Parâmetro	Descrição
Autenticação	<p>Opções de autenticação.</p> <p>None - nenhuma autenticação é usada.</p> <p>Basic - autenticação básica é usada.</p> <p>NTLM - autenticação NTLM (Windows NT LAN Manager) é usada.</p> <p>Kerberos - autenticação Kerberos é usada. Veja também: Configurando Kerberos com Zabbix.</p> <p>Digest - autenticação Digest é usada.</p> <p>Selecionar um método de autenticação proverá dois campos adicionais para informar nome de usuário e senha.</p> <p>Macros de usuário podem ser usadas nos campos de usuário e senha.</p>
Verificação de par SSL	<p>Marque a caixa de seleção para verificar o certificado do SSL do servidor web.</p> <p>O certificado do servidor será automaticamente obtido da localização da autoridade de certificado do sistema (CA). Você pode sobrescrever a localização dos arquivos CA usando o parâmetro de configuração SSLCALocation do Zabbix Server ou Proxy.</p> <p>Isto configura a opção cURL CURLOPT_SSL_VERIFYPEER.</p>
Verificação de host SSL	<p>Marque a caixa de seleção para verificar se o campo Nome comum (Common Name) ou o Nome de Assunto Alternativo (Subject Alternate Name) do certificado do servidor web corresponde.</p> <p>Isto define a opção cURL CURLOPT_SSL_VERIFYHOST.</p>
Arquivo de certificado SSL	<p>Nome do arquivo de certificado SSL usado para autenticação do cliente. O arquivo de certificado deve estar no formato PEM¹. Se o arquivo de certificado contém também a chave primária, deixe o campo Arquivo de chave SSL vazio. Se a chave está criptografada, especifique a senha no campo Senha da chave SSL. O diretório contendo este arquivo está especificado pelo parâmetro de configuração SSLCertLocation do Zabbix Server ou Proxy.</p> <p>Macros HOST.* e macros de usuário podem ser usadas neste campo.</p> <p>Isto define a opção cURL CURLOPT_SSLCERT.</p>

Parâmetro	Descrição
Arquivo de chave SSL	Nome do arquivo de chave privada SSL usado para autenticação de cliente. O arquivo de chave privada deve estar no formato PEM ¹ . O diretório contendo este arquivo está especificado pelo parâmetro de configuração SSLKeyLocation do Zabbix Server ou Proxy. Macros <code>HOST.*</code> e macros de usuário podem ser usadas neste campo.
Senha da chave SSL	Isto define a opção cURL <code>CURLOPT_SSLKEY</code> . Senha do arquivo de chave privada SSL. Macros de usuário podem ser usadas neste campo. Isto define a opção cURL <code>CURLOPT_KEYPASSWD</code> .

Attention:

[1] O Zabbix suporta arquivos de certificado e chave privada apenas no formato PEM. No caso de você ter seu dados de certificado e chave privada em arquivo no formato PKCS #12 (usualmente com extensão *.p12 ou *.pfx) você pode gerar o arquivo PEM a partir deles usando os seguintes comandos:

```
openssl pkcs12 -in ssl-cert.p12 -clcerts -nokeys -out ssl-cert.pem
openssl pkcs12 -in ssl-cert.p12 -nocerts -nodes -out ssl-cert.key
```

Note:

O Zabbix Server reconhece mudanças nos certificados sem uma reinicialização.

Note:

Se você possui o certificado do cliente e chave privada em um único arquivo, o especifique apenas em um campo "Arquivo de certificado SSL" e deixe o campo

"Arquivo de chave SSL" vazio. O certificado e chave ainda devem estar no formato PEM. Combinar certificado e chave é simples:

```
cat client.crt client.key > client.pem
```

Exibição Para visualizar cenários web configurados para um host, vá até Monitoramento → Hosts, localize o host na lista e clique no hyperlink Web na última coluna. Clique no nome do cenário para obter informações detalhadas.



Uma visão geral dos cenários web também pode ser exibida em Monitoramento → Dashboard por meio de um widget de Monitoramento web.

Resultados recentes da execução do cenário web estão disponíveis na seção Monitoramento → Últimos dados.

Monitoramento estendido Algumas vezes é necessário registrar em log o conteúdo recebido de uma página HTML. Isto é especialmente útil se algum cenário web falhar. O nível de depuração 5 (trace) server a esse propósito. Este nível pode ser definido nos arquivos de configuração do **Server** e **Proxy** ou usando uma opção de controle de tempo de execução (-R log_level_increase="http poller,N", onde N é o número do processo). Os seguintes exemplos demonstram como o monitoramento estendido pode ser iniciado considerando que o nível de depuração 4 já esteja definido:

Aumente o nível de log de todos o pollers http:

```
shell> zabbix_server -R log_level_increase="http poller"
```

Aumente o nível de log do segundo poller http:

```
shell> zabbix_server -R log_level_increase="http poller,2"
```

Se o monitoramento web estendido não é necessário ele pode ser parado usando a opção -R log_level_decrease.

1 Itens de monitoramento web

Visão geral

Alguns novos itens são automaticamente adicionados ao monitoramento quando cenários web são criados.

Todos os itens herdam etiquetas do cenário web.

Itens do cenário

Assim que um cenário é criado, o Zabbix automaticamente adiciona os seguintes itens para monitoramento.

Item	Descrição
Velocidade de download para cenário <cenário>	Este item coletará informações sobre a velocidade de download (bytes por segundo) de todo o cenário, p.e. média para todas as etapas. Chave de item: web.test.in[Cenário,,bps] Tipo: Numérico(float)
Etapas falha do cenário <cenário>	Este item exibirá o número da etapa que falhou no cenário. Se todas as etapas são executadas sucessivamente, 0 é retornado. Chave de item: web.test.fail[Cenário] Tipo: Numérico(unsigned)
Última mensagem de erro do cenário <cenário>	Este item retorna a última mensagem de texto de erro do cenário. Um novo valor é armazenado apenas se o cenário tem uma etapa falha. Se todas as etapas estão ok, nenhum novo valor é coletado. Chave de item: web.test.error[Cenário] Tipo: Caracter

O nome do cenário atual será usado em vez de "Cenário".

Note:

Itens de monitoramento web são adicionados com um período de retenção de 30 dias de histórico e 90 dias de tendência.

Note:

Se o nome de cenário começa com uma aspa dupla ou contém vírgula ou colchete, ele será apropriadamente quotado nas chaves de item. Em outros casos o nome não será quotado.

Estes itens podem ser usados para criar gatilhos e definir condições de notificação.

Exemplo 1

Para criar um gatilho "Cenário web falhou", você pode definir uma expressão de gatilho:

```
last(/host/web.test.fail[Cenário])>0
```

Certifique-se de substituir 'Cenário' com o nome real de seu cenário.

Exemplo 2

Para criar um gatilho "Cenário web falhou" com uma descrição de problema útil no nome do gatilho, você pode definir um gatilho com nome:

Cenário web "Cenário" falhou: {ITEM.VALUE}

e expressão de gatilho:

```
length(last(/host/web.test.error[Cenário]))>0 and last(/host/web.test.fail[Cenário])>0
```

Certifique-se de substituir 'Cenário' com o nome real de seu cenário.

Exemplo 3

Para criar um gatilho "Aplicação web está lenta", você pode definir uma expressão de gatilho:

```
last(/host/web.test.in[Cenário,,bps])<10000
```

Certifique-se de substituir 'Cenário' com o nome real de seu cenário.

Itens de etapa de cenário

Assim que uma etapa é criada, o Zabbix automaticamente adiciona os seguintes itens para monitoramento.

Item	Descrição
Velocidade de download para a etapa <Etapa> do cenário <Cenário>	Este item coletará informações sobre a velocidade de download (bytes por segundo) da etapa. Chave de item: web.test.in[Cenário,Step,bps] Tipo: Numérico(float)
Tempo de resposta para a etapa <Etapa> do cenário <Cenário>	Este item coletará informações sobre o tempo de resposta da etapa em segundos. O tempo de resposta é contado do início da requisição até todas as informações terem sido transferidas. Chave de item: web.test.time[Cenário,Step,resp] Tipo: Numérico(float)
Código de resposta para a etapa <Etapa> do cenário <Cenário>	Este item coletará códigos de resposta da etapa. Chave de item: web.test.rspcode[Cenário,Step] Tipo: Numérico(unsigned)

O cenário atual e nomes de etapa serão usados em vez de "Cenário" e "Etapa" respectivamente.

Note:

Os itens de monitoramento web são adicionados com um período de retenção de 30 dias para histórico e 90 dias para tendência.

Note:

Se o nome do cenário iniciar com uma aspa dupla ou contém vírgula ou colchete, ele será apropriadamente quotado na chaves de item. Em outros casos ele não será quotado.

Estes itens podem ser usados para criar gatilhos e definir condições de notificação. Por exemplo, para criar um gatilho "Autenticação Zabbix GUI está muito lenta", você pode definir uma expressão de gatilho:

```
last(/zabbix/web.test.time[ZABBIX GUI,Login,resp])>3
```

2 Cenário real de Monitoramento Web

Visão Geral

Esta seção apresenta um passo a passo de um exemplo real de como o Monitoramento Web pode ser usado.

Vamos utilizar o Monitoramento Web do Zabbix para monitorar a interface Web do Zabbix. Nós queremos saber se ela está disponível, se fornece o conteúdo correto e como está sua performance. Para isso nós precisamos nos autenticar com nosso usuário e senha.

Cenário

Etapa 1

Adicione um novo cenário Web

Nós adicionaremos um cenário para monitorar a interface Web do Zabbix. O cenário irá executar uma série de passos.

Vá para Configuração → Hosts, selecione um host e clique em Web na linha do host. Depois clique em Criar cenário Web.

Scenario
Steps
Tags
Authentication

*
Name

Zabbix frontend

*
Update interval

1m

*
Attempts

1

Agent

Zabbix

HTTP proxy

[protocol://][user[:password]@]proxy.example.com[:port]

Variables

Name		Value	
{password}	⇒	zabbix	Remove
{user}	⇒	Admin	Remove
Add			

Headers

Name		Value	
name	⇒	value	Remove
Add			

Enabled

☒

Add

Cancel

Todos os campos obrigatórios estão marcados com um asterisco vermelho.

O novo cenário Web nós iremos chamar de Zabbix frontend. Também criaremos duas variáveis: {user} e {password}.

Você também pode adicionar uma nova tag Application:Zabbix frontend na aba Etiquetas.

Etapa 2

Defina os passos do cenário.

Clique no botão Adicionar na aba Passos para adicionar cada passo.

Cenário Web Passo 01

Nós iniciaremos testando se a primeira página responde corretamente. Ela deverá retornar o código HTTP 200 e conter o texto "Zabbix SIA".

Step of web scenario

Name

First page

URL

http://localhost/zabbix/index.php

Parse

Query fields

Name

Value

name

⇒

value

Remove

Add

Post type

Form data

Raw data

Post fields

Name

Value

name

⇒

value

Remove

Add

Variables

Name

Value

name

⇒

value

Remove

Add

Headers

Name

Value

name

⇒

value

Remove

Add

Follow redirects

☒

Retrieve mode

Body

Headers

Body and headers

Timeout

15s

Required string

Zabbix SIA

Required status codes

200

Update

Cancel

Quando concluir a configuração do passo, clique em Adicionar.

Cenário Web Passo 02

Nós continuaremos autenticando no na interface Web do Zabbix e fazemos isso reutilizando as macros (variáveis) que definimos no nível do cenário - {user} e {password}.

Step of web scenario

* Name
Log in

* URL
http://localhost/zabbix/index.php
Parse

Query fields

Name	Value	
name	⇒	value Remove
Add		

Post type
Form data Raw data

Post fields

Name	Value	
name	⇒	{user} Remove
password	⇒	{password} Remove
enter	⇒	Sign in Remove
Add		

Variables

Name	Value	
{sid}	⇒	regex:name="csrf-token" content="([0- Remove
Add		

Headers

Name	Value	
name	⇒	value Remove
Add		

Follow redirects
☒

Retrieve mode
Body Headers Body and headers

* Timeout
15s

Required string

Required status codes
200

Update Cancel

::: Nota Importante Observe que a interface web do Zabbix usa redirecionamento JavaScript ao fazer o login, portanto, primeiro nós devemos fazer o login e somente em passos posteriores nós pode verificar os recursos logados. Além disso, a etapa de login deve ter a URL completa para o arquivo **index.php** . :::

Observe também como estamos obtendo o conteúdo da variável {sid} (session ID) usando uma sintaxe de variável com expressão regular: `regex:name="csrf-token" content="([0-9a-z]{16})"`. Essa variável será necessária no passo 4.

Cenário Web Passo 03

Estando logado, agora nós podemos verificar de fato. Para isso, verificamos uma string que só é visível quando conectado - por exemplo, **Administração**.

Step of web scenario

* Name
Login check

* URL
http://localhost/zabbix/index.php
Parse

Query fields

Name
Value

name
⇒
value
Remove

Add

Post type

Form data
Raw data

Post fields

Name
Value

name
⇒
value
Remove

Add

Variables

Name
Value

name
⇒
value
Remove

Add

Headers

Name
Value

name
⇒
value
Remove

Add

Follow redirects
☒

Retrieve mode

Body
Headers
Body and headers

* Timeout
15s

Required string
Administration

Required status codes
200

Update
Cancel

Cenário Web Passo 04

Agora que verificamos que o frontend está acessível, podemos fazer login e recuperar o conteúdo conectado, também devemos sair - caso contrário o banco de dados Zabbix ficará poluído com muitos e muitos registro de sessões abertas.

Step of web scenario

* Name

Log out

* URL

http://localhost/zabbix/index.php

Parse

Query fields

Name	Value	
sid	{sid}	Remove
reconnect	1	Remove
Add		

Post type

Form data
Raw data

Post fields

Name	Value	
name	value	Remove
Add		

Variables

Name	Value	
name	value	Remove
Add		

Headers

Name	Value	
name	value	Remove
Add		

Follow redirects

☒

Retrieve mode

Body
Headers
Body and headers

* Timeout

15s

Required string

Required status codes

200

Update

Cancel

Cenário Web Passo 05

Também podemos verificar se desconectamos procurando a string **Username** .

Step of web scenario

* Name

Logout check

* URL

http://localhost/zabbix/index.php

Parse

Query fields

Name

Value

name

⇒

value

Remove

Add

Post type

Form data

Raw data

Post fields

Name

Value

name

⇒

value

Remove

Add

Variables

Name

Value

name

⇒

value

Remove

Add

Headers

Name

Value

name

⇒

value

Remove

Add

Follow redirects

☒

Retrieve mode

Body

Headers

Body and headers

* Timeout

15s

Required string

Username

Required status codes

200

Update

Cancel

Configuração completa dos passos

Uma configuração completa dos passos do cenário da Web deve ter esta aparência:

Scenario

Steps 5

Tags 1

Authentication

* Steps

	Name	Timeout	URL	Required	Status
1:	First page	15s	http://localhost/zabbix/index.php	Zabbix SIA	200
2:	Log in	15s	http://localhost/zabbix/index.php		200
3:	Login check	15s	http://localhost/zabbix/index.php	Administration	200
4:	Log out	15s	http://localhost/zabbix/index.php		200
5:	Logout check	15s	http://localhost/zabbix/index.php	Username	200
Add					

Etapas 3

Salve o cenário de Monitoramento Web concluído.

O cenário será adicionado a um host. Para visualizar informações do cenário da web vá para Monitoramento → Hosts, localize o host na lista e clique em Web na última coluna.

Web monitoring					
Host	Name	Number of steps	Last check	Status	Tags
New host	Zabbix frontend	5	46s	OK	Application: Zabbix fro...
Displaying 1 of 1 found					

Clique no nome do cenário para ver estatísticas mais detalhadas:



10. Monitoramento de máquina virtual

Overview Suporte de monitoramento de ambientes VMware está disponível no Zabbix começando com a versão 2.2.0.

O Zabbix pode usar regras de descoberta de baixo nível para descobrir automaticamente Hypervisors VMware e máquinas virtuais e criar hosts para monitorar eles, com base em protótipos de host predefinidos.

O conjunto de dados padrão no Zabbix oferece vários modelos prontos para uso para monitorar o hypervisor VMware vCenter ou ESX.

A versão mínima necessária do VMware vCenter ou vSphere é 5.1.

Detalhes O monitoramento da máquina virtual é feito em duas etapas. Primeiro, virtuais os dados da máquina são coletados pelos processos do vmware collector Zabbix. Aqueles processos obtêm as informações necessárias dos serviços da Web da VMware no Protocolo SOAP, pré-processa e armazena no servidor Zabbix compartilhando memória. Em seguida, esses dados são recuperados pelos pesquisadores usando o Zabbix simples verifique **VMware keys**.

A partir do Zabbix versão 2.4.4 os dados coletados são divididos em 2 tipos: dados de configuração do VMware e dados do contador de desempenho do VMware. Ambos os tipos são coletados independentemente por coletores de vmware. Porque disso é recomendado habilitar mais coletores do que os monitorados Serviços VMware. Caso contrário, recuperação do contador de

desempenho VMware as estatísticas podem ser atrasadas pela recuperação da configuração do VMware dados (o que demora um pouco para grandes instalações).

Atualmente, apenas estatísticas de armazenamento de dados, interface de rede e dispositivo de disco e os itens do contador de desempenho personalizado são baseados no desempenho do VMware informações do contador.

Configuração Para que o monitoramento de máquinas virtuais funcione, o Zabbix deve ser **compilado** com o `--with-libxml2` e `--with-libcurl` opções de compilação.

As seguintes opções de arquivo de configuração podem ser usadas para ajustar o Virtual monitoramento da máquina:

- **StartVMwareCollectors** - o número de vmware pré-bifurcados instâncias do coletor.
Esse valor depende do número de serviços VMware que você vai monitor. Na maioria dos casos, deve ser:
 $\text{servicenum} < \text{StartVMwareCollectors} < (\text{servicenum} * 2)$
onde servicenum é o número de serviços VMware. Por exemplo. se vocês tem 1 serviço VMware para monitorar defina StartVMwareCollectors para 2, se você tem 3 serviços VMware, defina-o como 5. Observe que na maioria dos casos este valor não deve ser inferior a 2 e não deve ser 2 vezes maior que o número de serviços VMware que você monitora. Também tenha em mente que esse valor também depende do seu ambiente VMware tamanho e configuração VMwareFrequency e VMwarePerfFrequency parâmetros (veja abaixo).
- **VMwareCacheSize**
- **VMwareFrequency**
- **VMwarePerfFrequency**
- **VMwareTimeout**

Para mais detalhes, veja as páginas do arquivo de configuração do Zabbix **servidor** e **proxy**.

::: não importante Para dar suporte às métricas de capacidade do armazenamento de dados Zabbix requer que o parâmetro `vpxd.stats.maxQueryMetrics` de configuração VMware seja pelo menos 64. Consulte também a base de conhecimento VMware [artigo](#). :::

Descoberta O Zabbix pode usar uma regra de descoberta de baixo nível para descobrir automaticamente Hipervisores e máquinas virtuais VMware.

Discovery rule
Preprocessing
LLD macros
Filters
Overrides

* Name

Discover VMware hypervisors

Type

Simple check

* Key

vmware.hv.discovery[{\$VMWARE.URL}]

* Host interface

192.0.2.255:10050

User name

{\$VMWARE.USERNAME}

Password

{\$VMWARE.PASSWORD}

* Update interval

1h

Custom intervals

Type	Interval	Period
Flexible	Scheduling	50s
1-7,00:00		

Add

* Keep lost resources period

30d

Description

Discovery of hypervisors.

Enabled

☒

Add

Test

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

A chave da regra de descoberta na captura de tela acima é `vmware.hv.discovery[{$URL}]`.

Protótipos de host Os protótipos de host podem ser criados com a regra de descoberta de baixo nível. Quando máquinas virtuais são descobertas, esses protótipos se tornam hosts reais. Os protótipos, antes de serem descobertos, não podem ter seus próprios itens e gatilhos, além daqueles dos modelos vinculados. Anfitriões descobertos pertencerá a um host existente.

Discovery rules

All templates / VMware					
Items 3 Triggers Graphs Dashboards Discovery rules 4 Web scenarios					
<input type="checkbox"/> Template	Name ▲	Items	Triggers	Graphs	Hosts
<input type="checkbox"/> VMware	Discover VMware clusters	Item prototypes 1	Trigger prototypes	Graph prototypes	Host prototypes
<input type="checkbox"/> VMware	Discover VMware datastores	Item prototypes 4	Trigger prototypes	Graph prototypes	Host prototypes
<input type="checkbox"/> VMware	Discover VMware hypervisors	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes 1
<input type="checkbox"/> VMware	Discover VMware VMs	Item prototypes	Trigger prototypes	Graph prototypes	Host prototypes 1

Para que os hosts criados a partir de um protótipo tenham nomes de host exclusivos, o campo Nome do host deve conter pelo menos um **descoberta de baixo nível macro**.

Desde o Zabbix 5.2, os hosts descobertos podem ser configurados com interfaces ou herdar o IP de um host ao qual a regra de

descoberta pertence (predefinição). Para adicionar uma ou mais interfaces personalizadas, alterne a Interface seletor do modo Herdar para o modo Personalizado e pressione [Add](#) e selecione o tipo de interface necessário no menu suspenso que aparece. Todo tipos de interfaces suportados podem ser definidos para um protótipo de host: Agente Zabbix, SNMP, JMX, IPMI. Campos de interface suportam baixo nível macros de descoberta e **macros de usuário**. Se várias interfaces personalizadas forem especificadas - use a coluna Default para especifique a interface primária.

Notas:

- Se Personalizado for selecionado, mas nenhuma interface tiver sido especificada, o hosts serão criados sem interfaces.
- Se Herdar for selecionado para um protótipo de host que pertence a um template, os hosts descobertos herdarão a interface de um host para ao qual o modelo está vinculado.

Um host não será criado, se a interface do host contém dados incorretos

Host
IPMI
Tags
Macros
Inventory
Encryption

* Host name

Visible name

Templates

* Groups

Group prototypes

Add

Interfaces

InheritCustom

Type	IP address	DNS name
Agent	<input type="text" value="198.51.100.0"/>	<input type="text"/>
Agent	<input type="text"/>	<input type="text" value="{#LLDMACRO}"/>

Add

Monitored by proxy

(no proxy)

Create enabled
☒

Discover
☒

As macros LLD também podem ser usadas para o nome visível, protótipo do grupo de host campos, valores de tag ou valores de macros de usuário de protótipo de host.

Outras opções que podem ser especificadas para um protótipo de host são:

- Vinculação a grupos de hosts existentes
- Vinculação de modelos
- Criptografia

Se Criar habilitado estiver marcado, o host será adicionado em um Estado. Se desmarcado, o host será adicionado, mas em um estado desabilitado.

Se Discover estiver marcado (padrão), o host será criado. Se desmarcada, o host não será criado, a menos que esta configuração seja substituído na **descoberta regra**. este funcionalidade fornece flexibilidade adicional ao criar descoberta as regras.

Os hosts descobertos são prefixados com o nome da regra de descoberta que os criou, na lista de hosts. Hosts descobertos podem ser manualmente excluído. Os hosts descobertos também serão excluídos automaticamente, com base em o valor Manter o período de recursos perdidos (em dias) da regra de descoberta. A maioria das opções de configuração são somente leitura, exceto habilitar/desabilitar o host e o inventário do host. Os hosts descobertos não podem têm seus próprios protótipos de host.

Modelos prontos para uso O conjunto de dados padrão no Zabbix oferece vários modelos prontos para uso para monitorando o VMware vCenter ou diretamente o hipervisor ESX. Esses modelos contêm regras LLD pré-configuradas, bem como várias verificações integradas para monitorar instalações virtuais.

Modelos para monitoramento de hypervisor VMware vCenter e ESX:

- VMware - usa dados UUID para macros correspondentes;
- VMware FQDN - usa dados FQDN para macros correspondentes.

Note:

Para que o modelo VMware FQDN funcione corretamente cada VM monitorada deve ter um nome de SO exclusivo compatível com as regras de FQDN e o VMware Tools deve ser instalado em todas as máquinas. Se essas condições forem atendidos, é recomendável usar o modelo VMware FQDN. A criação do modelo VMware FQDN tornou-se possível após a introdução da capacidade para criar hosts com interfaces personalizadas no Zabbix 5.2.

Um modelo clássico VMware ainda está disponível e pode ser usado se o FQDN requisitos não podem ser atendidos. Lembre-se de que o VMware modelo tem um problema conhecido. Os hosts para máquinas virtuais descobertas ser criado com os nomes salvos no vCenter (por exemplo, VM1, VM2, etc.). Se o Zabbix agent active for instalado posteriormente nesses hosts com registro automático habilitado, o processo de registro automático lerá host nomes como eles foram registrados no lançamento (por exemplo, vm1.example.com, vm2.example.com, etc.) e crie novos hosts desde que não correspondências de nome foram encontradas. Como resultado, haverá duas duplicatas hosts para cada máquina com nomes diferentes.

Modelos usados pela descoberta (normalmente, esses modelos não devem ser vinculado manualmente a um host):

- VMware Hypervisor;
- VMware Convidado.

Templates

<input type="checkbox"/> Name ▲	Hosts	Items	Triggers	Graphs	Dashboards	Discovery	Web
<input type="checkbox"/> VMware	Hosts	Items 3	Triggers	Graphs	Dashboards	Discovery 4	Web
<input type="checkbox"/> VMware FQDN	Hosts	Items 3	Triggers	Graphs	Dashboards	Discovery 4	Web
<input type="checkbox"/> VMware Guest	Hosts	Items 27	Triggers 1	Graphs	Dashboards	Discovery 3	Web
<input type="checkbox"/> VMware Hypervisor	Hosts	Items 26	Triggers 4	Graphs	Dashboards	Discovery 2	Web
<input type="checkbox"/> VMWare SD-WAN VeloCloud by HTTP	Hosts	Items 7	Triggers 5	Graphs	Dashboards	Discovery 5	Web

Configuração do host Para usar verificações simples do VMware, o host deve ter as seguintes macros de usuário definidas - **{ \$VMWARE.URL }** - SDK de serviço VMware (vCenter ou hipervisor ESX) URL (<https://servername/sdk>) - **{ \$VMWARE.USERNAME }** - Nome de usuário do serviço VMware - **{ \$VMWARE.PASSWORD }** - Serviço VMware { \$VMWARE.USERNAME } usuário senha

Exemplo O exemplo a seguir demonstra como configurar rapidamente o VMware Monitorando no Zabbix:

- compilar o servidor zabbix com as opções necessárias (--with-libxml2 e --com-libcurl)
- defina a opção StartVMwareCollectors na configuração do servidor Zabbix arquivo para 1 ou mais
- criar um novo host
- defina as macros de host necessárias para autenticação VMware:

```
{{...:assets:en:manual:vm_monitoring:vm_host_macros.png|}}
```

* Vincule o host ao modelo de serviço VMware:

```
{{...:assets:en:manual:vm_monitoring:vm_host_templates.png|}}
```

* Clique no botão //Adicionar// para salvar o host

Registro estendido Os dados coletados pelo coletor VMware podem ser registrados para depuração usando o nível de depuração 5. Este nível pode ser definido em **servidor** e **proxy** arquivos de configuração ou usando uma opção de controle de tempo de execução (-R log_level_increase="vmware collector,N", onde N é um processo número). Os exemplos a seguir demonstram como o log estendido pode ser iniciado desde que o nível de depuração 4 já esteja definido:

```
Aumente o nível de log de todos os coletores de vmware:
shell> zabbix_server -R log_level_increase="vmware collector"
```

```
Aumente o nível de log do segundo coletor vmware:
shell> zabbix_server -R log_level_increase="vmware collector,2"
```

Se o registro estendido dos dados do coletor VMware não for necessário, pode ser que parou de usar a opção -R log_level_decrease.

Solução de problemas

- Em caso de métricas indisponíveis, verifique se não estão indisponível ou desativado por padrão no VMware vSphere recente versões ou se alguns limites não são colocados na métrica de desempenho consultas de banco de dados. Ver [ZBX-12094](#) para detalhes adicionais.
- No caso de 'config.vpxd.stats.maxQueryMetrics' ser inválido ou excede o número máximo de caracteres permitido** erro, adicione um parâmetro config.vpxd.stats.maxQueryMetrics para o vCenter Server definições. O valor deste parâmetro deve ser o mesmo que o valor de maxQuerysize no web.xml da VMware. Veja este VMware base de conhecimento [artigo](#) para detalhes.

1 Campos de chave de descoberta de máquina virtual

A seguinte tabela mostra os campos retornados pelas chaves de descoberta relacionadas à maquina virtual.

Chave de item		
Descrição	Campo	Conteúdo recuperado
vmware.cluster.discovery Realiza descoberta de cluster.	{ #CLUSTER.ID } { #CLUSTER.NAME }	Identificador do cluster. Nome do cluster.
vmware.datastore.discovery Realiza descoberta de datastore.	{ #DATASTORE.NAME } { #DATASTORE.OBJECT }	Nome do datastore. Objeto JSON com um array de {instance-Name:partitionId}.
vmware.dc.discovery Realiza descoberta de datacenter.	{ #DATACENTER.ID } { #DATACENTER.ID }	Nome do datacenter. ID do datacenter.
vmware.hv.discovery Realiza descoberta de hypervisor.	{ #HV.UUID }	Identificador de hypervisor único.

Chave de item	
	{#HV.ID} Identificador de hypervisor (Nome de objeto gerenciado por HostSystem).
	{#HV.NAME} Nome do hypervisor.
	{#HV.NETNAME} Nome do host de rede hypervisor.

Chave de item

{#HV.IP} Endereço IP do hypervisor, pode estar vazio. No caso de uma configuração HA com múltiplas interfaces de rede, a seguinte seleção de prioridade para interface é observada:

- favorece o IP que compartilha a subrede com o IP do vCenter
- favorece o IP da subrede com gateway padrão
- favorece o IP da interface com menor ID

Este campo é suportado desde o Zabbix 5.2.2.

{#CLUSTER.NAME}do cluster, pode estar vazio.

{#DATACENTER.NAME}datacenter.

Chave de item

		<p><code>{#PARENT.NAME}</code> Nome do container que armazena o hypervisor. Suportado desde o Zabbix 4.0.3.</p> <p><code>{#PARENT.TYPE}</code> Tipo de container no qual o hypervisor é armazenado. Os valores podem ser Datacenter, Folder, ClusterComputerResource, VMware, onde 'VMware' representa um tipo de container desconhecido. Suportado desde o Zabbix 4.0.3.</p>
vmware.hv.datastore.discovery	Realiza descoberta de datastore de hypervisor. Note que múltiplos hypervisors podem usar o mesmo datastore.	<p><code>{#DATASTORE.NAME}</code> Nome do datastore.</p> <p><code>{#MULTIPATH.LOCATION}</code> Número de caminhos de datastore registrados.</p> <p><code>{#MULTIPATH.PARTITION.COUNT}</code> Número de partições de disco disponíveis.</p>
vmware.vm.discovery	Realiza descoberta de máquina virtual.	<p><code>{#VM.UUID}</code> Identificador de máquina virtual único.</p>

Chave de item

{#VM.ID}	Identificador de máquina virtual (Nome de objeto gerenciado pela Virtual-Machine).
{#VM.NAME}	Nome da máquina virtual.
{#HV.NAME}	Nome do hypervisor.
{#DATACENTER.NAME}	Nome do datacenter.
{#CLUSTER.NAME}	Nome do cluster, pode estar vazio.
{#VM.IP}	Endereço IP da máquina virtual, pode estar vazio. Suportado desde o Zabbix 5.2.2.
{#VM.DNS}	Nome DNS da máquina virtual, pode estar vazio. Suportado desde o Zabbix 5.2.2.
{#VM.GUEST.FAMILY}	Família do SO hospede da máquina virtual, pode estar vazio. Suportado desde o Zabbix 5.2.2.

Chave de item

		{#VM.GUESTFOLDERTITLENAME} com- pleto do SO hóspede da máquina virtual, pode estar vazio. Suportado desde o Zabbix 5.2.2.
		{#VM.FOLDERNAME} Pode de pastas pai da máquina virtual, pode ser usado como valor para grupos anin- hados; nomes de pasta são com- binados com "/". Pode estar vazio. Suportado desde o Zabbix 5.4.2.
vmware.vm.net.if.discovery	Realiza descoberta de interface de rede da máquina virtual.	{#IFNAME} Nome da interface de rede.
vmware.vm.vfs.dev.discovery	Realiza descoberta de dispositivo de disco da máquina virtual.	{#DISKNAME} Nome do disposi- tivo de disco.
vmware.vm.vfs.fs.discovery	Realiza descoberta de sistema de arquivo da máquina virtual.	{#FSNAME} Nome do sistema de arquivo.

11. Manutenção

Visão geral Você pode definir períodos de manutenção para grupos de host, hosts e gatilhos/serviços no Zabbix.

Há dois tipos de manutenção - com coleta de dados e sem coleta de dados.

Durante uma manutenção "com coleta de dados" os gatilhos são processados como de costume e eventos são criados quando necessário. No entanto, as escalasões de problema são pausadas para os hosts/gatilhos em manutenção, se a opção Pausar operações para problemas suprimidos estiver marcada na configuração da ação. Neste caso, as etapas de escalação que podem incluir o envio de notificações ou comandos remotos serão ignoradas pela duração do período de manutenção. Note que as operações de recuperação de atualização de problema não são suprimidas durante a manutenção, apenas as escalasões.

Por exemplo, se as etapas de escalação são programadas para 0, 30 e 60 minutos após o início de um problema, e há uma manutenção com duração de meia hora ocorrendo entre 10 minutos e 40 minutos após o surgimento do problema real, as etapas dois e três serão executadas meia hora mais tarde, ou 60 minutos e 90 minutos (considerando que o problema ainda exista). Da mesma forma, se um problema surge durante a manutenção, a escalação iniciará após a manutenção.

Para receber notificações de problema durante a manutenção normalmente (sem atraso), você tem que desmarcar a opção Pausar operações para problemas suprimidos na configuração da ação.

Note:

Se ao menos um host (usado na expressão de gatilho) não está em modo de manutenção, o Zabbix enviará uma notificação de problema.

O Zabbix Server deve estar em execução durante a manutenção. Processos timer são responsáveis pela alteração de estado do host de/para manutenção aos 0 segundos de cada minuto. Note que quando um host entra em manutenção, os processos timer do Zabbix Server lerão todos os problemas abertos para verificar se é necessário suprimi-los. Isto pode ter um impacto de performance se houver muitos problemas abertos. O Zabbix Server também lerá todos os problemas abertos ao iniciar, mesmo que não haja manutenções configuradas no momento.

Um Proxy sempre coletará dados independentemente do tipo de manutenção (incluindo manutenção "sem dados"). Os dados são posteriormente ignorados pelo Server se 'sem coleta de dados' estiver definido.

Quando uma manutenção "sem dados" termina, os gatilhos usando a função `nodata()` não dispararão antes da próxima verificação durante o período que estão verificando.

Se um item de log é adicionado enquanto um host está em manutenção e a manutenção termina, apenas as novas entradas de log desde o fim da manutenção serão reunidas.

Se um valor de registro de data é enviado para um host que está em um tipo de manutenção "sem dados" (p.e. usando **Zabbix sender**) então este valor será eliminado, no entanto é possível enviar um valor de registro de data para um período de manutenção expirado e ele será aceito.

Attention:

Para garantir um comportamento previsível de períodos de manutenção recorrentes (diário, semanal, mensal), é necessário usar um timezone comum para todas as partes do Zabbix.

Se um período de manutenção, hosts, grupos ou etiquetas são alterados pelo usuário, as mudanças somente terão efeito após a sincronização do cache de configuração.

Attention:

When creating a maintenance period, the **time zone** of the user who creates it is used. However, when recurring maintenance periods (Daily, Weekly, Monthly) are scheduled, the time zone of the Zabbix server is used. To ensure predictable behavior of recurring maintenance periods, it is required to use a common time zone for all parts of Zabbix.

Configuração Para configurar um período de manutenção:

- Vá até: Configuração → Manutenção
- Clique em Criar período de manutenção (ou no nome de um período de manutenção existente)
- Informe os parâmetros de manutenção no formulário

* Name

Server regular maintenance

Maintenance type

With data collection

No data collection

* Active since

2021-01-01 00:00

* Active till

2022-01-01 00:00

* Periods

Period type	Schedule	Period	Action
One time only	2020-04-17 11:33	1y	Edit Remove
Add			

Host groups

type here to search

Hosts

Server2 X

type here to search

* At least one host group or host must be selected.

Tags

And/Or

Or

tag

Contains

Equals

value

[Remove](#)

[Add](#)

Description

We break and fix things at this time.

Add

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	Nome do período de manutenção.
Tipo de manutenção	Dois tipos de de manutenção podem ser definidos: Com coleta de dados - os dados serão coletados pelo Server durante a manutenção, gatilhos serão processados Sem coleta de dados - os dados não serão coletados pelo Server durante a manutenção
Ativo a partir de	A data e hora quando a execução dos períodos de manutenção se torna ativa. Nota: A definição deste horário somente não ativa um período de manutenção; para isso vá até a aba Períodos.
Ativo até	A data e hora quando a execução dos períodos de manutenção deixa de estar ativo.

Parâmetro	Descrição
Períodos	Este bloco permite que você defina os dias e horas exatas quando a manutenção ocorre. Clicar em Add abre uma janela com um formulário flexível de Período de manutenção onde você podem definir uma programação de manutenção. Consulte Períodos de manutenção para uma descrição detalhada.
Grupos de host	Selecione os grupos de host para os quais a manutenção será ativada. A manutenção será ativada para todos os hosts do(s) grupo(s) de hosts especificado. Este campo é autocompletável, então ao iniciar a digitar nele uma lista suspensa com todos os grupos de host será exibida. Especificar um grupo de host pai implicitamente seleciona todos os grupos de host aninhados. Desta forma a manutenção também será ativada para nos hosts dos grupos aninhados.
Hosts	Selecione os hosts para os quais a manutenção será ativada. Este campo é autocompletável, então ao iniciar a digitar nele uma lista suspensa com todos os grupos de host será exibida.
Etiquetas	Se etiquetas de manutenção são especificadas, a manutenção para os hosts selecionados ainda serão ativadas, mas problemas serão apenas suprimidos (p.e. nenhuma ação será tomada) se suas etiquetas correspondem. No caso de múltiplas etiquetas, elas serão calculadas como segue: And/Or - todas as etiquetas devem corresponder; no entanto etiquetas com o mesmo nome de etiqueta são calculadas pela condição Or Or - suficiente se uma etiqueta corresponde Há dois modos de correspondência do valor da etiqueta: Contains - correspondência de substring sensível a maiúsculas e minúsculas (valor da etiqueta contém a string informada) Equals - correspondência de string sensível a maiúsculas e minúsculas (valor da etiqueta é igual à string informada)
Descrição	Descrição do período de manutenção.

Períodos de manutenção

A janela de período de manutenção serve para planejamento de horário para uma manutenção recorrente ou pontual. O formulário é dinâmico com os campos disponíveis mudando com base no Tipo de período selecionado.

Maintenance period

Period type

One time only

* Date

2020-04-17 11:33

* Maintenance period length

365

Days

0

Hours

0

Minutes

Apply

Cancel

Tipo de período	Descrição
Apenas uma vez	Define a data e hora, e a duração do período de manutenção.

Tipo de período	Descrição
Diário	<p>Todo(s os) dia(s) - frequência de manutenção: 1 (padrão) - todo dia, 2 - a cada dois dias, etc.</p> <p>Às (hora:minuto) - momento do dia no qual a manutenção inicia</p> <p>Duração do período de manutenção - por quanto tempo a manutenção ficará ativa.</p>
Semanal	<p>Toda(s as) semana(s) - frequência de manutenção: 1 (padrão) - todo dia, 2 - a cada dois dias, etc.</p> <p>Dia da semana - dia no qual a manutenção deve ocorrer.</p> <p>Às (hora:minuto) - momento do dia no qual a manutenção inicia</p> <p>Duração do período de manutenção - por quanto tempo a manutenção ficará ativa.</p>
Mensal	<p>Mês - selecione todos os meses durante os quais a manutenção regular é executada.</p> <p>Data: Dia do mês - Selecione esta opção se a manutenção ocorre na mesma data a cada mês (por exemplo, todo 1º dia do mês). Então, selecione o dia requerido no novo campo que aparece.</p> <p>Data: Dia da semana - Selecione esta opção se a manutenção ocorre apenas em certos dias (por exemplo, toda primeira segunda-feira do mês). Então, na lista suspensa selecione a semana requerida do mês (primeira, segunda, terceira, quarta, ou última) e marque a caixas de seleção para o(s) dia(s) de manutenção.</p> <p>Às (hora:minuto) - momento do dia no qual a manutenção inicia</p> <p>Duração do período de manutenção - por quanto tempo a manutenção ficará ativa.</p>

Quando pronto, pressione Adicionar para adicionar o período de manutenção ao bloco de Períodos.

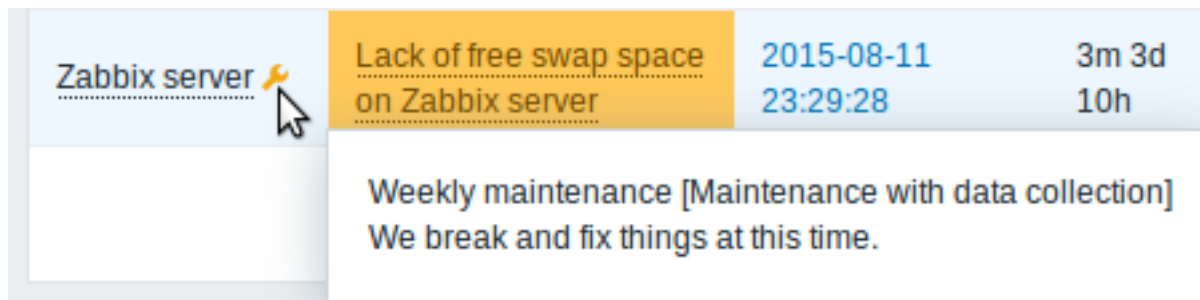
Notas:

- Quando o parâmetro Todo dia/Toda semana é maior que 1, o dia ou semana de início é o dia/semana no qual o horário de Ativo a partir de se encontra. Por exemplo:
 - com Ativo a partir de definido para 1 de Janeiro às 12h e uma manutenção de uma hora para cada dois dias às 23h resultará no primeiro período de manutenção iniciando em 1 de Janeiro às 23h, enquanto o segundo período de manutenção iniciará em 3 de Janeiro às 23h;
 - com a mesma data de Ativo a partir de e uma manutenção de uma hora definida para cada dois dias à 1h, o primeiro período de manutenção iniciará em 3 de Janeiro à 1h, enquanto o segundo período de manutenção iniciará em 5 de Janeiro à 1h.
- Mudanças de Horário de Verão (Daylight Saving Time - **DST**) não afetam quão longa será a manutenção. Digamos que tenhamos uma manutenção de duas horas que usualmente inicia à 1h e finaliza às 3h:
 - Se após uma hora de manutenção (às 2h) uma alteração de Horário de Verão ocorrer e o horário atual mudar de 2h para 3h, a manutenção continuará por mais uma hora até às 4h;
 - Se após duas horas de manutenção (às 3h) uma alteração de Horário de Verão ocorrer e o horário atual mudar de 3h para 2h, a manutenção parará porque duas horas se passaram.

Exibição Exibindo hosts em manutenção

Um ícone de chave inglesa laranja  próximo ao nome do host indica que este host está em manutenção:

- Monitoramento → Dashboard
- Monitoramento → Problemas
- Inventário → Hosts → Detalhes de inventário de host
- Configuração → Hosts (Veja a coluna 'Estado (status)')



Detalhes de manutenção são exibidos quando o ponteiro do mouse é posicionado sobre o ícone.

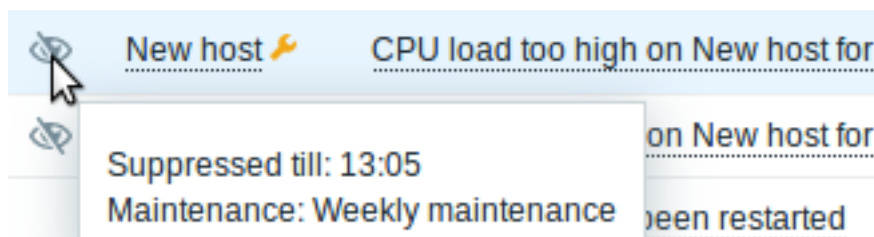
Adicionalmente, os hosts em manutenção recebem um fundo laranja em Monitoramento → Mapas.

Exibindo problemas suprimidos

Normalmente os problemas para os hosts em manutenção são suprimidos, ou seja não exibidos no Frontend. No entanto, também é possível configurar que os problemas suprimidos sejam mostrados, pela seleção da opção **Mostrar problemas suprimidos** nestes locais:

- Monitoramento → Dashboard (na configuração do widget Hosts com problema, Problemas, Problemas por severidade, Visão geral de gatilho)
- Monitoramento → Problemas (no filtro)
- Monitoramento → Mapas (na configuração de mapa)
- **Notificações** globais (na configuração de perfil de usuário)

Quando problemas suprimidos são exibidos, o seguinte ícone é mostrado: . Mover um mouse sobre o ícone exibe mais detalhes:



12. Expressões regulares

Visão geral [Expressões regulares compatíveis com Perl](#) (PCRE, PCRE2) são suportadas no Zabbix.

Há dois modos de usar expressões regulares no Zabbix:

- informando manualmente uma expressão regular
- usando uma expressão regular global criada no Zabbix

Expressões regulares Você pode informar manualmente uma expressão regular nos locais suportados. Note que a expressão pode não começar com @ porque esse símbolo é usado no Zabbix para referenciar expressões regulares globais.

Warning:

É possível ficar sem stack quando usando expressões regulares. Consulte a [página de manual pcrestack](#) para mais informações.

Note que em correspondência multilinear, as âncoras ^ e \$ correspondem ao início/fim de cada linha respectivamente, em vez do início/fim da string completa.

Expressões regulares globais Há um editor avançado para criação e teste de expressões regulares complexas no Zabbix Frontend.

Uma vez que uma expressão regular é criada desta forma, ela pode ser usada em vários locais no Frontend referenciando seu nome, prefixado com @, por exemplo, @meuregcustomizado.

Para criar uma expressão regular global:

- Vá até: Administração → Geral

- Selecione Expressões regulares da lista suspensa
- Clique em Nova expressão regular

A aba **Expressões** permite definir o nome da expressão regular e adicionar subexpressões.

Expressions

Test

* Name

Network interfaces for discovery

* Expressions

Expression type	Expression	Delimiter	Case s
Result is FALSE	^Software Loopback Interface		<input checked="" type="checkbox"/>
Result is FALSE	^(In)?[Ll]oop[Bb]ack[0-9._]*\$		<input checked="" type="checkbox"/>
Result is FALSE	^NULL[0-9.]*\$		<input checked="" type="checkbox"/>
Result is FALSE	^[Ll]o[0-9.]*\$		<input checked="" type="checkbox"/>
Result is FALSE	^[Ss]ystem\$		<input checked="" type="checkbox"/>
Result is FALSE	^Nu[0-9.]*\$		<input checked="" type="checkbox"/>

Add

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	Defina o nome da expressão regular. Quaisquer caracteres Unicode são permitidos.
Expressões	<p>Clique em Adicionar no bloco Expressões para adicionar uma nova subexpressão.</p> <p>Tipo de expressão:</p> <p>String de caractere inclusa - corresponde a substring qualquer substring de uma lista delimitada. A lista delimitada inclui uma vírgula (,), um ponto (.) ou uma barra (/).</p> <p>String de caractere não inclusa - corresponde qualquer string exceto a substring</p> <p>Resultado é VERDADEIRO - corresponde a expressão regular</p> <p>Resultado é FALSO - não corresponde à expressão regular</p> <p>Informe substring/expressão regular.</p>
Delimitador	Uma vírgula (,), um ponto (.) ou uma barra (/) para separar strings de texto em uma expressão regular. Este parâmetro está ativo apenas quando o tipo de expressão "Qualquer string de caractere inclusa" estiver selecionado.
Sensível a maiúsculas/minúsculas	Uma caixa de seleção para especificar se uma expressão regular é sensível à capitalização de letras.

Uma barra (/) na expressão é tratada literalmente, em vez de um delimitador. Desta forma é possível salvar expressões contendo uma barra, sem erros.

Attention:

Um nome de expressão regular customizada no Zabbix pode conter vírgulas, espaços, etc. Nos casos onde isto pode incorrer em má interpretação quando referenciando (por exemplo, uma vírgula no parâmetro de uma chave de item) a referência completo pode ser colocada entre quotas da seguinte forma: "@Meu regex customizado para objetivo1, objetivo2".

Nomes de expressão regular não devem ser quotados em outros locais (por exemplo, em propriedades de regra LLD).

Na aba **Teste** a expressão regular e suas subexpressões pode ser testadas pelo fornecimento de uma string de teste.

Expressions

Test

Test string

lo

Test expressions

Result

Expression type	Expression	Result
Result is FALSE	^Software Loopback Interface	TRUE
Result is FALSE	^(In)?[Ll]oop[Bb]ack[0-9._]*\$	TRUE
Result is FALSE	^NULL[0-9.]*\$	TRUE
Result is FALSE	^[Ll]o[0-9.]*\$	FALSE
Result is FALSE	^[Ss]ystem\$	TRUE
Result is FALSE	^Nu[0-9.]*\$	TRUE
Combined result		FALSE

Os resultados mostram o estado de cada subexpressão e o estado da expressão customizada total.

O estado de expressão customizada total é definido como Resultado combinado. Se várias subexpressões são definidas o Zabbix usa o operador lógico AND para calcular o Resultado combinado. Isto significa que se ao menor um Resultado for Falso, o Resultado combinado também terá um estado Falso.

Expressões regulares globais padrão O Zabbix vem com várias expressões regulares globais em seu conjunto de dados padrão.

Nome	Expressão	Correspondências
Descoberta de sistemas de arquivo	^(btrfs ext2 ext3 ext4 jfs reiser xfs ufs fat32 fat16 fat12 fat1 fat vfat ntfs hfs refs apfs zfs)\$	Strings começando com "btrfs" ou "ext2" ou "ext3" ou "ext4" ou "jfs" ou "reiser" ou "xfs" ou "ufs" ou "fat32" ou "fat16" ou "fat12" ou "fat1" ou "vfat" ou "ntfs" ou "hfs" ou "refs" ou "apfs" ou "zfs"
Interfaces de rede para descoberta	^Software Loopback Interface	Strings começando com "Software Loopback Interface".
	^lo\$	"lo"
	^(In)?[Ll]oop[Bb]ack[0-9._]*\$	Strings que opcionalmente começam com "In", e então têm "L" ou "l", então "oop", então "B" ou "b", então "ack", que pode ser opcionalmente seguido por qualquer número de dígitos, pontos ou sublinhas.
	^NULL[0-9.]*\$	Strings começando com "NULL" opcionalmente seguidas por qualquer número de dígitos ou pontos.

Nome	Expressão	Correspondências
	<code>^[Ll]o[0-9.]*\$</code>	Strings começando com "Lo" ou "lo" e opcionalmente seguidas por qualquer número de dígitos ou pontos.
	<code>^[Ss]ystem\$</code>	"System" ou "system"
	<code>^[Nu][0-9.]*\$</code>	Strings começando com "Nu" opcionalmente seguidas por qualquer número de dígitos ou pontos.
Dispositivos de armazenamento para descoberta SNMP	<code>^(Physical memory\\ Virtual memory\\ Memory buffers\\ Cached memory\\ Swap space)\$</code>	"Physical memory (Memória física)" ou "Virtual memory (Memória virtual)" ou "Memory buffers (Buffers de memória)" ou "Cached memory (Memória em cache)" ou "Swap space (Memória de troca)"
Nomes de serviço Windows para descoberta	<code>^(MMCSS\\ gupdate\\ SysmonLog\\ clr_optimization_v2.0.50727_32\\ clr_optimization_v4.0.30319_32\\ gupdate\$</code>	"MMCSS" ou strings como "SysmonLog" ou strings como "clr_optimization_v2.0.50727_32" e "clr_optimization_v4.0.30319_32" onde em vez de pontos você pode colocar qualquer caractere exceto nova linha.
Estado de início de serviço Windows para descoberta	<code>^(automatic\\ automatic delayed)\$</code>	"automatic (automático)" ou "automatic delayed (automático com atraso)"

Exemplos Exemplo 1

Uso da seguinte expressão em descoberta de baixo-nível para descobrir bancos de dados, exceto um banco de dados com um nome específico:

`^TESTDATABASE$`

Test string

TESTDATABASE

Test expressions

Result	Expression type	Expression	Result
	Result is FALSE	<code>^TESTDATABASE</code>	FALSE
	Combined result		FALSE

Tipo de expressão escolhido: "Resultado é FALSO". Não corresponde ao nome, contendo string "TESTDATABASE".

Exemplo com um modificador regex em linha

Uso da seguinte expressão regular incluindo um modificador em linha (?i) para corresponder os caracteres "erro":

`(?i)erro`

Test string

Sometexthere1345Error1357

Test expressions

Result	Expression type	Expression	Result
	Result is TRUE	(?i)error	TRUE
	Combined result		TRUE

Tipo de expressão escolhido: "Resultado é VERDADEIRO". Os caracteres "erro" são correspondidos.

Outro exemplo com um modificador regex em linha

Uso da seguinte expressão regular incluindo múltiplos modificadores em linha para corresponder aos caracteres após uma linha específica:

(?<=testar (?i)tudo(?-i) apos esta linha\n)(?sx).*# nós adicionamos o modificador s para permitir . corres

Test string

Some text here for your consideration
1235kfd345
match eveRything after this line
Continuation

Test expressions

Result	Expression type	Expression	Result
	Result is TRUE	(?<=match (?i)everything(?-i) after this line\n)(?sx).*# we add s modifier to allow . match newline characters	TRUE
	Combined result		TRUE

Tipo de expressão escolhido: "Resultado é VERDADEIRO". Os caracteres após uma linha específica são correspondidos.

Attention:
 O modificador **g** não pode ser especificado em linha. A lista de modificadores disponíveis pode ser encontrada na [página de manual pcreyntax](#). Para mais informações sobre a sintaxe PCRE, por favor consulte a [documentação HTML PCRE](#).

Suporte a expressão regular por localização

Localização	Expressão regular	Expressão regular global	Comentários
Itens do Agent			
	everylog[]	Sim	parâmetros regexp, severity, source, eventId
	log[]		parâmetro regexp
	log.count[]		

Localização	Expressão regular	Expressão regular global	Comentários
	logrt[]	Sim/Não	parâmetro regex suporta ambos, parâmetro file_regex suporta apenas expressões não-globais
	logrt.count[] proc.cpu.util[]	Não	parâmetro cmdline
	proc.mem[] proc.num[] sensor[]		parâmetros device e sensor no Linux 2.4
	system.hw.macaddr[]		parâmetro interface
	system.sw.packages[]		parâmetro package
	vfs.dir.count[]		parâmetros regex_incl, regex_excl, regex_excl_dir
	vfs.dir.size[]		parâmetros regex_incl, regex_excl, regex_excl_dir
	vfs.file.regex[]		parâmetro regex
	vfs.file.regmatch[] web.page.regex[]		
SNMP traps	snmptrap[]	Sim	parâmetro regex
Pré- processamento de valor de item Funções para itens de gatilho/calculados	Sim	Não	parâmetro pattern
	count(\$m)	Sim	parâmetro pattern se parâmetro operator é regex ou iregex
	countunique()	Sim	
	find(\$m)	Sim	
	logeventid()	Sim	parâmetro pattern
	logsource()		
Descoberta de baixo-nível			

Localização	Expressão regular	Expressão regular global	Comentários
	Filtres (fil- tra)	Sim	Campo Expressão regular
	Overrides (so- bre- screve)	Não	Nas opções corresponde (matches), não corresponde (does not match) para condições
Condições de ação	Sim	Não	Operação Nas opções corresponde (matches), não corresponde (does not match) para condições de autoregistro Nome do host (Host name) e Metadados do host (Host metadata)
Monitoramento web	Sim	Não	Variáveis com um prefixo regex: Campo String requerida
Contexto de macro de usuário	Sim	Não	Em contexto de macro com um prefixo regex:
Funções de macro	regsbr()	Não	parâmetro pattern
	iregsub()		
Mapeamento de ícone	Sim	Sim	Campo Expressão
Mapeamento de valor	Sim	Não	Campo Valor se o tipo de mapeamento é regexp

13. Reconhecimento do problema

Visão geral Eventos de problema no Zabbix podem ser reconhecidos pelos usuários.

Se um usuário é notificado sobre um evento de problema, ele pode acessar o Zabbix Frontend, abrir a janela de atualização de problema usando uma das formas listadas abaixo e reconhecer o problema. Quando reconhecendo, ele pode inserir seu comentário para o evento, dizendo que está trabalhando nele ou o que mais ele achar necessário ser informado sobre o evento.

Deste modo, se outro usuário do sistema visualizar o mesmo problema, ele imediatamente vê se ele foi reconhecido e os comentários até o momento.

Desta forma, o processo de solução de problemas com mais de um usuário de sistema pode ocorrer de uma maneira coordenada.

O estado do reconhecimento também é usado quando definindo **operações de ação**. Você pode definir, por exemplo, que uma notificação seja enviada para um gestor de mais alto nível apenas se um evento não for reconhecido por um certo período de tempo.

Para reconhecer eventos e comentar neles, um usuário deve ter ao menos permissão de leitura nos gatilhos correspondentes. Para alterar a severidade ou encerrar o problema, um usuário deve ter permissões de leitura-escrita aos gatilhos correspondentes.

Há **várias** formas de acessar a janela de atualização de problema, o que permite reconhecer um problema.

- ·Você pode selecionar os problemas em Monitoramento → Problemas e então ·clique em Atualização em massa abaixo da lista
- ·Você pode clicar na coluna Ack mostrando o estado de reconhecimento dos ·problemas em: ··Monitoramento → Dashboard (widgets Problemas e Problemas por severidade) ··Monitoramento → Problemas ··Monitoramento → Problemas → Detalhes do evento

A coluna Ack contém um link 'Sim' ou um link 'Não', indicando um problema reconhecido ou não reconhecido respectivamente. Ao clicar nos links será apresentada a janela de atualização de problema.

- ·Você pode clicar na célula de um problema não resolvido em: ··Monitoramento → Dashboard (widget Visão geral do gatilho)

O menu dinâmico contém uma opção Reconhecer que o levará para a janela de atualização do problema.

Atualizando problemas O pop-up de atualização do problema permite:

- comente o problema
- ver comentários e ações até agora
- alterar a gravidade do problema
- reconhecer/desconhecer problema
- fechar manualmente o problema

Update problem

Problem **/:** Disk space is critically low (>90% used)

Message

History

Time	User	User action	Message
2020-05-07 11:27:50	Admin (Zabbix Administrator)	✕	
2020-05-07 11:27:43	Admin (Zabbix Administrator)	✓	Ok

Scope

☒ Only selected problem

☐ Selected and all other problems of related triggers 1 event

Change severity

☐ Not classified

☐ Information

☐ Warning

☐ Average

☐ High

☐ Disaster

Acknowledge

☐

Close problem

☐

* At least one update operation or message must exist.

Update

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parameter	Description
Problem	Se apenas um problema for selecionado, o nome do problema será exibido.
Message	Se vários problemas forem selecionados, N problemas selecionados será exibido. Digite o texto para comentar o problema (máximo de 2048 caracteres).

Parameter	Description
History	Atividades anteriores e comentários sobre o problema são listados, juntamente com o horário e os detalhes do usuário. Para o significado dos ícones usados para denotar as ações do usuário, consulte a página detalhe do evento . Observe que o histórico é exibido se apenas um problema for selecionado para a atualização.
Scope	Defina o escopo de tais ações como alterar a gravidade, reconhecer ou fechar manualmente os problemas: Only selected problem - afetará apenas este evento Selected and all other problems of related triggers - em caso de problema de confirmação/fechamento, afetará este evento e todos os outros problemas que não foram confirmados/fechados até o momento. Se o escopo contiver problemas já reconhecidos ou encerrados, esses problemas não serão reconhecidos/fechados repetidamente. Por outro lado, o número de operações de mudança de mensagem e gravidade não é limitado.
Change severity	Marque a caixa de seleção e clique no botão de gravidade para atualizar a gravidade do problema. A caixa de seleção para alterar a gravidade estará disponível se existirem permissões de leitura e gravação para pelo menos um dos problemas selecionados. Somente os problemas que podem ser lidos serão atualizados ao clicar em Update. Se as permissões de leitura e gravação não existirem para nenhum dos gatilhos selecionados, a caixa de seleção será desativada.
Acknowledge	Marque a caixa de seleção para reconhecer o problema. Esta caixa de seleção está disponível se houver pelo menos um problema não confirmado entre os selecionados. Não é possível adicionar outra confirmação para um problema já reconhecido (mas é possível adicionar outro comentário).
Unacknowledge	Marque a caixa de seleção para cancelar o reconhecimento do problema. Esta caixa de seleção estará disponível se houver pelo menos um problema reconhecido entre os selecionados.
Close problem	Marque a caixa de seleção para fechar manualmente o(s) problema(s) selecionado(s). A caixa de seleção para fechar um problema está disponível se a opção Allow manual close estiver marcado em configuração da trigger para pelo menos um dos problemas selecionados. Serão fechados apenas os problemas que podem ser fechados ao clicar em Update. Se nenhum problema puder ser fechado manualmente, a caixa de seleção será desativada. Problemas já fechados não serão fechados repetidamente.

Exibição Com base nas informações de confirmação é possível configurar como a contagem de problemas é exibida no painel ou nos mapas. Para fazer isso, você tem que fazer seleções na opção Problem display, disponível em Ambas [map configuration](#) e Problems by severity [dashboard widget](#). É possível exibir toda a contagem de problemas, problemas não reconhecidos contar como separado da contagem total ou não reconhecida do problema.

Com base nas informações de atualização do problema (confirmação, etc.), é possível configurar operações de atualização - enviar uma mensagem ou executar comandos remotos.

14. Configuração exportação/importação

Visão geral A funcionalidade de exportação/importação Zabbix torna possível a troca de várias entidades de configuração entre um sistema Zabbix e outro.

Casos típicos de utilização para esta funcionalidade:

- partilha de modelos ou mapas de rede - os utilizadores Zabbix podem partilhar os seus parâmetros de configuração
- partilhar cenários web em share.zabbix.com - exportar um modelo com os cenários web e carregar para share.zabbix.com. Depois, outros podem descarregar o modelo e importar o ficheiro para o Zabbix.
- integrar com ferramentas de terceiros - YAML universal, XML e JSON formatos tornam possível a integração e a importação/exportação de dados com terceiros ferramentas e aplicações do partido

O que pode ser exportado/importado

Os objectos que podem ser exportados/importados são:

- **grupos anfitriões** (* através do Zabbix API apenas*)
- **templates**
- **hospedeiros**
- **mapas de rede**
- **tipos de mídia**
- **imagens**

Formato de exportação

Os dados podem ser exportados usando o frontend da web Zabbix ou **Zabbix API**. Os formatos de exportação suportados são YAML, XML e JSON.

Detalhes sobre exportação

- Todos os elementos suportados são exportados num único ficheiro.
- Entidades hospedeiras e modelos (itens, gatilhos, gráficos, descoberta regras) que são herdados de modelos ligados não são exportados. Quaisquer alterações feitas a essas entidades a nível de anfitrião (tais como alterações intervalo de itens, expressão regular modificada ou adição de protótipos a a regra da descoberta de baixo nível) será perdida quando exportar; quando importação, todas as entidades de modelos ligados são recriadas como em o modelo original ligado.
- Entidades criadas por descoberta de baixo nível e quaisquer entidades dependentes sobre eles não são exportados. Por exemplo, um gatilho criado para um O artigo gerado pela regra LLD não será exportado.

Detalhes sobre importação

- A importação irá parar no primeiro erro.
- Ao atualizar imagens existentes durante a importação de imagens, campo "imagetype" é ignorado, ou seja, é impossível alterar o tipo de imagem via importação.
- Ao importar hosts/modelos usando a opção "Excluir ausente", macros de host/modelo não presentes no arquivo de importação serão excluídas também.
- Tags vazias para itens, gatilhos, gráficos, aplicativos de host/modelo, DiscoveryRules, itemPrototypes, triggerPrototypes, graphPrototypes são sem sentido, ou seja, é o mesmo que se estivesse faltando. Outras marcas, por exemplo, aplicativos de itens, são significativos, ou seja, tags vazias significam sem aplicativos para item, falta de tag significa não atualizar formulários.
- Importação suporta YAML, XML e JSON, o arquivo de importação deve ter um extensão de arquivo correta: .yaml e .yml para YAML, .xml para XML e .json para JSON.
- Consulte **informações de compatibilidade** sobre as versões XML suportadas.

```
zabbix_export:  
  versão: '6.0'  
  data: '2020-04-22T06:20:11Z'
```

Formato base YAML

```
zabbix_export:
```

Nó raiz para exportação Zabbix YAML.

```
  versão: '6.0'
```

Versão de exportação.

```
  data: '2020-04-22T06:20:11Z'
```

Data em que a exportação foi criada no formato longo ISO 8601.

Outros nós são dependentes de objetos exportados.

```
<?xml version="1.0" encoding="UTF-8"?>  
<zabbix_export>  
  <version>6.0</version>  
  <date>2020-04-22T06:20:11Z</date>  
</zabbix_export>
```

Formato XML

```
<?xml version="1.0" encoding="UTF-8"?>
```

Cabeçalho padrão para documentos XML.

```
<zabbix_export>
```

Elemento raiz para exportação XML do Zabbix.

```
<version>6.0</version>
```

Versão de exportação.

```
<date>2020-04-22T06:20:11Z</date>
```

Data em que a exportação foi criada no formato longo ISO 8601.

Outras tags são dependentes de objetos exportados.

```
{
  "zabbix_export": {
    "versão": "6.0",
    "data": "2020-04-22T06:20:11Z"
  }
}
```

Formato JSON

```
"zabbix_export":
```

Nó raiz para exportação Zabbix JSON.

```
"versão": "6.0"
```

Versão de exportação.

```
"data": "2020-04-22T06:20:11Z"
```

Data em que a exportação foi criada no formato longo ISO 8601.

Outros nós são dependentes de objetos exportados.

1 Grupos de anfitriões

Nos grupos de host front-end podem ser **exportados** apenas com exportação de host ou modelo. Quando um host ou modelo é exportado todos os grupos a que pertence são exportados com ele automaticamente.

A API permite exportar grupos de hosts independentemente de hosts ou modelos.

Formato de exportação

```
grupos:
-
  nome: 'servidores Zabbix'
```

grupos/grupo

Parâmetro	Tipo	Descrição	Detalhes
name	string	Nome do grupo.	

2 Templates

COMMENT https://git.zabbix.com/projects/WEB/repos/documentation/browse/en/manual/xml_export_import/templates.md?at=refs%2Fheads

CONTEXT manual/xml_export_import/templates.md RESOURCE Documentation 6.0•manual/xml_export_import/templates.xliff

Visão geral

Os modelos são **exportados** com muitos objetos e relações objetais.

A exportação de modelo contém:

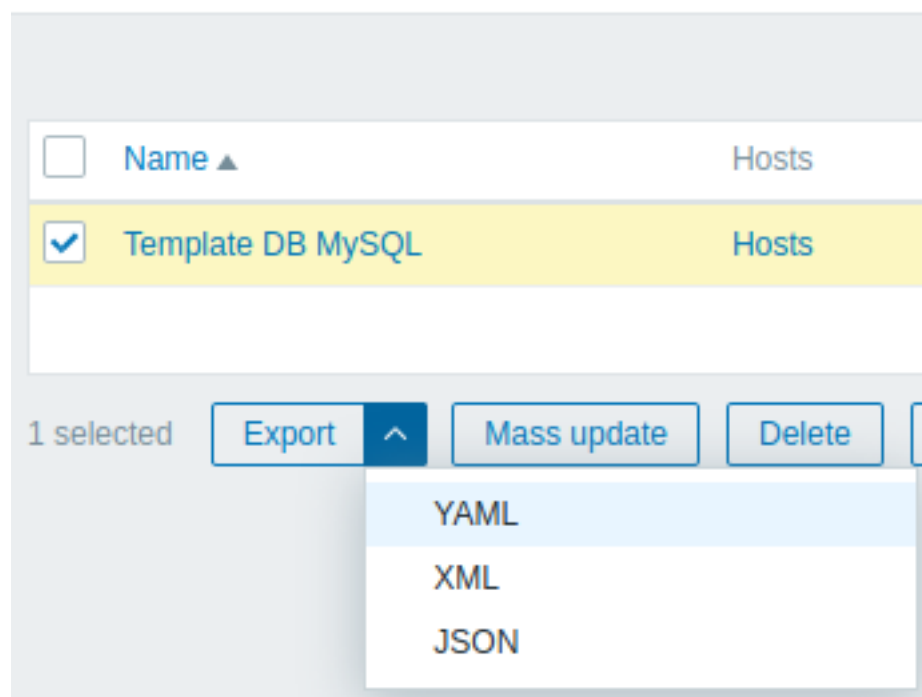
- grupos de hosts vinculados
- dados do modelo
- ligação a outros modelos
- ligação a grupos de acolhimento
- itens diretamente vinculados
- gatilhos diretamente ligados
- gráficos diretamente ligados
- painéis diretamente vinculados
- regras de descoberta diretamente vinculadas com todos os protótipos
- cenários da web diretamente vinculados
- mapas de valor

Exportando

Para exportar modelos, faça o seguinte:

- Vá para: Configuração → Modelos
- Marque as caixas de seleção dos modelos para exportar
- Clique em Exportar abaixo da lista

≡ Templates



Dependendo do formato selecionado, os modelos são exportados para um arquivo local com um nome padrão:

- zabbix_export_templates.yaml - na exportação YAML (opção padrão Para exportar)
- zabbix_export_templates.xml - na exportação XML
- zabbix_export_templates.json - na exportação JSON

Importando

Para importar modelos, faça o seguinte:

- Vá para: Configuração → Modelos
- Clique em Importar à direita
- Selecione o arquivo de importação
- Marque as opções necessárias nas regras de importação
- Clique em Importar

Import

* Import file

Choose file

template_power_apc_ups_snmp.xml

Rules	Update existing	Create new	Delete missing
Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Value mappings	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template dashboards	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template linkage		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Web scenarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Import

Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Regras de importação:

Regra	Descrição
Atualizar existente	Os elementos existentes serão atualizados com os dados retirados do arquivo de importação. Caso contrário, eles não serão atualizados.
Criar novo	A importação adicionará novos elementos usando dados do arquivo de importação. Caso contrário, ele não irá adicioná-los.
Delete missing	A importação removerá elementos existentes não presentes no arquivo de importação. Caso contrário, ele não os removerá. Se Excluir ausente estiver marcado para vinculação de modelo, a vinculação de modelo existente não presente no arquivo de importação será removida do modelo junto com todas as entidades herdadas dos modelos potencialmente desvinculados (itens, gatilhos, etc).

Na próxima tela, você poderá visualizar o conteúdo de um modelo sendo importado. Se este for um novo modelo, todos os elementos serão listados em verde. Ao atualizar um modelo existente, novos elementos de modelo são destacado em verde; os elementos de template removidos são destacados em vermelho; os elementos que não foram alterados são listados em um plano de fundo cinza.

Templates

▼ Updated

▼ Templates

VMware

```
templates:
  template: VMware
- name: VMware
+ name: 'VMware alternative'
- description: "You can discuss this template or leave feedback on our forum"
+ description: "You can discuss this fabulous template or leave feedback on our forum"
groups:
  - name: Templates/Applications
tags:
  - tag: class
    value: software
  - tag: target
    value: vmware
macros:
  - macro: '${VMWARE.PASSWORD}'
    description: 'VMware service {USERNAME} user password'
  - macro: '${VMWARE.URL}'
    description: 'VMware service (vCenter or ESX hypervisor) SDK URL (https://...)'
  - macro: '${VMWARE.USERNAME}'
    description: 'VMware service user name'
```

O menu à esquerda pode ser usado para navegar pela lista de mudanças. A seção Atualizado destaca todas as alterações feitas em elementos do modelo. A seção Adicionado lista novos elementos de modelo. Os elementos em cada seção são agrupados por tipo de elemento; pressione o cinza seta para baixo para expandir ou recolher o grupo de elementos.

Templates

▼ Updated

▲ Templates

▲ Items

▲ Triggers

▲ Discovery rules

▲ Item prototypes

▼ Dashboards

UPS Summary

▼ Graphs

Capacity of the UPS batteries

▼ Added

▼ Items

Battery capacity

System name

Revise as alterações do modelo e pressione Importar para realizar a importação do modelo. Uma mensagem de sucesso ou falha da importação será exibida na parte dianteira.

Formato de exportação

Formato de exportação em YAML:

```
zabbix_export:
  version: '6.0'
  date: '2021-08-31T12:40:55Z'
  groups:
    - uuid: a571c0d144b14fd4a87a9d9b2aa9fcd6
      name: Templates/Applications
  templates:
    - uuid: 56079badd056419383cc26e6a4fcc7e0
      template: VMware
      name: VMware
      description: |
        You can discuss this template or leave feedback on our forum https://www.zabbix.com/forum/zabbix-s

        Template tooling version used: 0.38
  templates:
    - name: 'VMware macros'
  groups:
    - name: Templates/Applications
  items:
    - uuid: 5ce209f4d94f460488a74a92a52d92b1
      name: 'VMware: Event log'
      type: SIMPLE
      key: 'vmware.eventlog[{$VMWARE.URL},skip]'
      history: 7d
      trends: '0'
      value_type: LOG
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'Collect VMware event log. See also: https://www.zabbix.com/documentation/6.0/manual'
      tags:
        - tag: Application
          value: VMware
    - uuid: ee2edad8b8ce943ef81d25dbbba8667a4
      name: 'VMware: Full name'
      type: SIMPLE
      key: 'vmware.fullname[{$VMWARE.URL}]'
      delay: 1h
      history: 7d
      trends: '0'
      value_type: CHAR
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'VMware service full name.'
      preprocessing:
        - type: DISCARD_UNCHANGED_HEARTBEAT
          parameters:
            - 1d
      tags:
        - tag: Application
          value: VMware
    - uuid: a0ec9145f2234fba79a28c57ebdb44d
      name: 'VMware: Version'
      type: SIMPLE
      key: 'vmware.version[{$VMWARE.URL}]'
      delay: 1h
      history: 7d
      trends: '0'
      value_type: CHAR
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
```

```

description: 'VMware service version.'
preprocessing:
  - type: DISCARD_UNCHANGED_HEARTBEAT
    parameters:
      - 1d
tags:
  - tag: Application
    value: VMware
discovery_rules:
- uuid: 16ffc933cce74cf28a6edf306aa99782
  name: 'Discover VMware clusters'
  type: SIMPLE
  key: 'vmware.cluster.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Discovery of clusters'
  item_prototypes:
    - uuid: 46111f91dd564a459dbc1d396e2e6c76
      name: 'VMware: Status of "{#CLUSTER.NAME}" cluster'
      type: SIMPLE
      key: 'vmware.cluster.status[{$VMWARE.URL},{#CLUSTER.NAME}]'
      history: 7d
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'VMware cluster status.'
      valuemap:
        name: 'VMware status'
      tags:
        - tag: Application
          value: VMware
- uuid: 8fb6a45cbe074b0cb6df53758e2c6623
  name: 'Discover VMware datastores'
  type: SIMPLE
  key: 'vmware.datastore.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  item_prototypes:
    - uuid: 4b61838ba4c34e709b25081ae5b059b5
      name: 'VMware: Average read latency of the datastore {#DATASTORE}'
      type: SIMPLE
      key: 'vmware.datastore.read[{$VMWARE.URL},{#DATASTORE},latency]'
      history: 7d
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'Amount of time for a read operation from the datastore (milliseconds).'
      tags:
        - tag: Application
          value: VMware
    - uuid: 5355c401dc244bc588ccd18767577c93
      name: 'VMware: Free space on datastore {#DATASTORE} (percentage)'
      type: SIMPLE
      key: 'vmware.datastore.size[{$VMWARE.URL},{#DATASTORE},pfree]'
      delay: 5m
      history: 7d
      value_type: FLOAT
      units: '%'
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'VMware datastore space in percentage from total.'
      tags:

```

```

    - tag: Application
      value: VMware
- uuid: 84f13c4fde2d4a17baaf0c8c1eb4f2c0
  name: 'VMware: Total size of datastore {#DATASTORE}'
  type: SIMPLE
  key: 'vmware.datastore.size[{$VMWARE.URL},{#DATASTORE}]'
  delay: 5m
  history: 7d
  units: B
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'VMware datastore space in bytes.'
  tags:
    - tag: Application
      value: VMware
- uuid: 540cd0fbc56c4b8ea19f2ff5839ce00d
  name: 'VMware: Average write latency of the datastore {#DATASTORE}'
  type: SIMPLE
  key: 'vmware.datastore.write[{$VMWARE.URL},{#DATASTORE},latency]'
  history: 7d
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Amount of time for a write operation to the datastore (milliseconds).'
  tags:
    - tag: Application
      value: VMware
- uuid: a5bc075e89f248e7b411d8f960897a08
  name: 'Discover VMware hypervisors'
  type: SIMPLE
  key: 'vmware.hv.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Discovery of hypervisors.'
  host_prototypes:
    - uuid: 051a1469d4d045cbbf818fcc843a352e
      host: '{#HV.UUID}'
      name: '{#HV.NAME}'
      group_links:
        - group:
            name: Templates/Applications
      group_prototypes:
        - name: '{#CLUSTER.NAME}'
        - name: '{#DATACENTER.NAME}'
      templates:
        - name: 'VMware Hypervisor'
      macros:
        - macro: '{$VMWARE.HV.UUID}'
          value: '{#HV.UUID}'
          description: 'UUID of hypervisor.'
      custom_interfaces: 'YES'
      interfaces:
        - ip: '{#HV.IP}'
- uuid: 9fd559f4e88c4677a1b874634dd686f5
  name: 'Discover VMware VMs'
  type: SIMPLE
  key: 'vmware.vm.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Discovery of guest virtual machines.'
  host_prototypes:

```

```

- uuid: 23b9ae9d6f33414880db1cb107115810
  host: '#{VM.UUID}'
  name: '#{VM.NAME}'
  group_links:
    - group:
        name: Templates/Applications
  group_prototypes:
    - name: '#{CLUSTER.NAME} (vm)'
    - name: '#{DATACENTER.NAME}/{#VM.FOLDER} (vm)'
    - name: '#{HV.NAME}'
  templates:
    - name: 'VMware Guest'
  macros:
    - macro: '${VMWARE.VM.UUID}'
      value: '#{VM.UUID}'
      description: 'UUID of guest virtual machine.'
  custom_interfaces: 'YES'
  interfaces:
    - ip: '#{VM.IP}'
valuemaps:
- uuid: 3c59c22905054d42ac4ee8b72fe5f270
  name: 'VMware status'
  mappings:
    - value: '0'
      newvalue: gray
    - value: '1'
      newvalue: green
    - value: '2'
      newvalue: yellow
    - value: '3'
      newvalue: red

```

Tags de elemento

Os valores de tag de elemento são explicados na tabela abaixo.

Tags de modelo

Elemento	Propriedade do elemento	Requerido	Tipo	Intervalo	Descrição																																																																		
templates	Elemento raiz para templates.	uuid	string	Identificador exclusivo para este modelo.	template	string	Nome do modelo exclusivo.	nome	string	Nome do modelo visível.	descricao	text	Descrição do modelo.	groups	x	Elemento raiz para grupos de hosts de modelo.	uuid	string	Identificador exclusivo para este grupo de hosts.	name	string	Nome do grupo de hosts.	templates	x	Elemento raiz para templates vinculados.	nome	string	Nome do modelo.	tags	x	Elemento raiz para tags de modelo.	tag	string	Nome da tag.	valor	string	Valor da tag.	macros	x	Elemento raiz para macros de usuário de modelo.	macro	string	Nome da macro do usuário.	type	string	0 - TEXT (padrão) 1 - SECRET_TEXT 2 - VAULT	Tipo da macro.	valor	string	Valor da macro do usuário.	descricao	string	Descrição da macro do usuário.	valuemaps	x	Elemento raiz para mapas de valor de modelo.	uuid	string	Identificador exclusivo para este mapa de valores.	nome	string	Nome do mapa de valores.	mapping	x	Elemento raiz para mapeamentos.	valor	string	Valor de um mapeamento.	newvalue	string	Novo valor de um mapeamento.

Tags de item de modelo

Elemento	Propriedade do elemento	Requerido	Tipo	Intervalo ¹	Descrição																										
items	Elemento raiz para itens.	uuid	string	Identificador exclusivo para o item.	nome	string	Nome do item.	type	string	0 - ZABBIX_PASSIVE (padrão) 2 - TRAP 3 - SIMPLE 5 - INTERNO 7 - ZABBIX_ACTIVE 10 - EXTERNO 11 - ODBC 12 - IPMI 13 - SSH 14 - TELNET 15 - CALCULADO 16 - JMX 17 - SNMP_TRAP 18 - DEPENDENT 19 - HTTP_AGENT 20 - SNMP_AGENT 21 - ITEM_TYPE_SCRIPT	Tipo de item.	snmp_oid	string	ID do objeto SNMP.	Requerido por itens SNMP.	chave	string	Chave do item.	delay	string	Padrão: 1m	Intervalo de atualização do item.	Aceita segundos ou uma unidade de tempo com sufixo (30s, 1m, 2h, 1d).	Intervalos personalizados podem ser especificados como intervalos flexíveis ou agendamento.	Vários intervalos são separados por um ponto e vírgula.	As macros do usuário podem ser usadas. Uma única macro deve preencher todo o campo. Múltiplas macros em um campo ou macros misturadas com texto não são suportadas.	Intervalos flexíveis podem ser escritos como duas macros separadas por uma barra (por exemplo, \${FLEX_INTERVAL}/\${FLEX_PERIOD}).	history	string	Padrão: 90d	Uma unidade de tempo de quanto tempo

os dados do histórico devem ser armazenados. Unidade de tempo com sufixo, macro de usuário ou macro LLD. |<|tendências|<string|Padrão: 365d|Uma unidade de tempo de quanto tempo os dados de tendências devem ser armazenados. Unidade de tempo com sufixo, macro de usuário ou macro LLD. |<|status|<string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do item. |<|value_type|<string|0 - FLOAT
1 - CHAR
2 - LOG
3 - UNSIGNED (padrão)
4 - TEXT|Tipo de valor recebido. |<|allowed_hosts|<string|<|Lista de endereços IP (delimitados por vírgula) de hosts com permissão para enviar dados para o item.

Usado por itens de agente de trapper e HTTP. |<|unidades|<string|<|Unidades de valores retornados (bps, B, etc.) |<|params|<text|<|Parâmetros adicionais dependendo do tipo do item:
- script executado para itens de Script, SSH e Telnet;
- consulta SQL para itens de monitor de banco de dados;
- fórmula para itens calculados. |<|ipmi_sensor|<string|<|Sensor IPMI.

Usado apenas por itens IPMI. |<|auth_type|<string|Tipo de autenticação para itens do agente SSH:
0 - SENHA (padrão)
1 - PUBLIC_KEY

Tipo de autenticação para itens do agente HTTP:
0 - NONE (padrão)
1 - BASIC
2 - NTLM|Tipo de autenticação.

Usado apenas por itens de agente SSH e HTTP. |<|username|<string|<|Nome de usuário para autenticação.
Usado por verificação simples, SSH, Telnet, monitor de banco de dados, itens de agente JMX e HTTP.

Requerido por SSH e Telnet items.
Quando usado pelo agente JMX, a senha também deve ser especificada junto com o nome de usuário ou ambas as propriedades devem ser deixadas em branco. |<|password|<string|<|Senha para autenticação.
Usada por verificação simples, SSH, Telnet, monitor de banco de dados, itens do agente JMX e HTTP.

Quando usado pelo agente JMX, o nome de usuário também deve ser especificado junto com a senha ou ambas as propriedades devem ser deixadas em branco. |<|publickey|<string|<|Nome do arquivo de chave pública.

Obrigatório para itens do agente SSH. |<|privatekey|<string|<|Nome do arquivo de chave privada.

Obrigatório para itens do agente SSH. |<|port|<string|<|Porta personalizada monitorada pelo item.
Pode conter macros de usuário.

Usado apenas por itens SNMP. |<|descricao|<text|<|Descrição do item. |<|inventory_link|<string|0 - NONE

Nome do campo de inventário do host em letras maiúsculas. Por exemplo:
4 - ALIAS
6 - OS_FULL
14 - HARDWARE
etc.|Campo de inventário do host que é preenchido pelo item.

Consulte o [host página de inventário](#) para obter uma lista de campos de inventário de host compatíveis e seus IDs. |<|logtimefmt|<string|<|Formato da hora nas entradas de log.
Usado apenas por itens de log. |<|jmx_endpoint|<string|<|JMX endpoint.

Usado apenas por itens do agente JMX. |<|url|<string|<|String de URL.

Requerido apenas para itens de agente HTTP. |<|allow_traps|<string|0 - NO (padrão)
1 - YES|Permitir preencher o valor como em um item de trapper.

Usado apenas por itens de agente HTTP. |<|follow_redirects|<string|0 - NÃO
1 - SIM (padrão)|Seguir redirecionamentos de resposta HTTP ao agrupar dados.

Usado apenas por itens de agente HTTP. |headers|<|<|<|Elemento raiz para cabeçalhos de solicitação HTTP(S), em que o nome do cabeçalho é usado como chave e o valor do cabeçalho como valor.
Usado apenas por itens de agente HTTP. |<|nome|x|<string|<|Nome do cabeçalho. |<|valor|x|<string|<|Valor do cabeçalho. |<|http_proxy|<string|<|String de conexão do proxy HTTP(S).

Usado apenas por itens do agente HTTP. |<|output_format|<string|0 - RAW (padrão)
1 - JSON|Como processar a resposta.

Usado apenas por itens do agente HTTP. |<|post_type|<string|0 - RAW (padrão)
2 - JSON
3 - XML|Tipo de corpo de dados da postagem.

Modelo de tags de regra de descoberta de baixo nível

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo|Descrição| |-----|-----|-----|-----|-----|-----| |discovery_rules|<|<|<|Elemento raiz para regras de descoberta de baixo nível. |<|Para a maioria dos valores de tag de elemento, veja valores de tag de elemento para um item regular. Somente as tags específicas para regras de descoberta de baixo nível são descritas abaixo.<|<|<|<|<|type|<string|0 - ZABBIX_PASSIVE (padrão)
2 - TRAP
3 - SIMPLE
5 - INTERNO
7 - ZABBIX_ACTIVE
10 - EXTERNO
11 - ODBC
12 - IPMI
13 - SSH
14 - TELNET
16 - JMX
18 - DEPENDENTE
19 - HTTP_AGENT
20 - SNMP_AGENT|Tipo de item. |<|vidal|<string|Padrão: 30d|Período de tempo após o qual os itens que não são mais descobertos serão excluídos. Segundos, unidade de tempo com sufixo ou macro de usuário. |filtro|<|<|<|<|Filtro individual. |<|evaltype|<string|0 - AND_OR (padrão)
1 - AND
2 - OU
3 - FORMULA|Lógica a ser usada para verificar o filtro de regra de descoberta de baixo nível condições. |<|formula|<string|<|Fórmula de cálculo personalizada para condições de filtro. |conditions|<|<|<|Elemento raiz para condições de filtro. |<|macro|x|<string|<|Nome da macro de descoberta de baixo nível. |<|valor|<string|<|Valor do filtro: expressão regular ou expressão regular global. |<|operador|<string|8 - MATCHES_REGEX (padrão)
9 - NOT_MATCHES_REGEX|Operador de condição. |<|formulaid|x|character|<|ID exclusivo arbitrário que é usado para fazer referência a uma condição da expressão personalizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente. |lld_macro_paths|<|<|<|Elemento raiz para caminhos de macro LLD. |<|lld_macro|x|<string|<|Nome da macro de descoberta de baixo nível. |<|path|x|<string|<|Seletor para o valor que será atribuído à macro correspondente. |pré-processamento|<|<|<|<|pré-processamento do valor da regra LLD. |step|<|<|<|<|Etapa de pré-processamento do valor da regra LLD individual. |<|Para a maioria dos valores de tag de elemento, consulte valores de tag de elemento para um pré-processamento de valor de item de modelo. Somente as tags que são específicas para o pré-processamento de valor de descoberta de baixo nível do modelo são descritas abaixo.<|<|<|<|<|type|x|<string|5 - REGEX
11 - XMLPATH
12 - JSONPATH
15 - NOT_MATCHES_REGEX
16 - CHECK_JSON_ERROR
17 - CHECK_XML_ERROR
20 - DISCARD_UNCHANGED_HEARTBEAT
21 - JAVASCRIPT
23 - PROMETHEUS_TO_JSON
24 - CSV_TO_JSON
25 - STR_REPLACE |Tipo da etapa de pré-processamento do valor do item. |trigger_prototypes|<|<|<|Elemento raiz para protótipos de trigger. |<|Para valores de tag de elemento de protótipo de acionador, consulte tags regulares de [gatilho de modelo](#).<|<|<|<|graph_prototypes|<|<|<|Elemento raiz para protótipos de gráfico. |<|Para valores de tag de elemento de protótipo de gráfico, consulte tags regulares [template graph](#).<|<|<|<|host_prototypes|<|<|<|Elemento raiz para protótipos de host. |<|Para valores de tag de elemento de protótipo de host, consulte tags regulares [host](#).<|<|<|<|item_prototypes|<|<|<|Elemento raiz para protótipos de itens. |<|Para valores de tag de elemento de protótipo de item, consulte tags regulares [template item](#).<|<|<|<|

|master_item|<|<|<|Dados de protótipo de item/protótipo de item de protótipo de item individual.|<|key|x|string|<|Valor da chave do protótipo do item mestre/protótipo do item dependente.

Obrigatório para um item dependente.

Tags de gatilho de modelo

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|-----|-----|-----|
|triggers|<|<|<|Elemento raiz para triggers.|<|uuid|x|string|<|Identificador exclusivo para este gatilho.|<|expressão|x|string|<|Expressão de gatilho.|<|recovery_mode|-|string|0 - EXPRESSION (padrão)
1 - RECOVERY_EXPRESSION
2 - NONE|Base para geração de eventos OK.|<|recuperação_expression|-|string|<|Expressão de recuperação de gatilho.|<|nome|x|string|<|Nome do gatilho.|<|correlation_mode|-|string|0 - DISABLED (padrão)
1 - TAG_VALUE|Modo de correlação (sem correlação de evento ou correlação de evento por tag).|<|correlation_tag|-|string|<|O nome do tag a ser usado para correlação de eventos.|<|url|-|string|<|URL associado ao gatilho.|<|status|-|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do gatilho.|<|prioridade|-|string|0 - NOT_CLASSIFIED (padrão)
1 - INFO
2 - AVISO
3 - MÉDIA
4 - ALTA
5 - DESASTRE |Gravidade do gatilho.|<|descrição|-|text|<|Descrição do gatilho.|<|type|-|string|0 - SINGLE (padrão)
1 - MULTIPLE|Tipo de geração de evento (evento de problema único ou vários eventos de problema).|<|manual_close|-|string|0 - NÃO (padrão)
1 - SIM|Fechamento manual de eventos de problema.|dependencies|<|<|<|Elemento raiz para dependências.|<|nome|x|string|<|Nome do gatilho de dependência.|<|expression|x|string|<|Expressão do gatilho de dependência.|<|recuperação_expression|-|string|<|Expressão de recuperação do gatilho de dependência.|tags|<|<|<|Elemento raiz para tags de acionamento.|<|tag|x|string|<|Nome da tag.|<|valor|-|string|<|Valor da tag.|

Tags de gráfico de modelo

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|-----|-----|-----|
|gráficos|<|<|<|Elemento raiz para gráficos.|<|uuid|x|string|<|Identificador único para este gráfico.|<|nome|x|string|<|Nome do gráfico.|<|width|-|integer|20-65535 (padrão: 900)|Largura do gráfico, em pixels. Usado para visualização e para gráficos de pizza/explodidos.|<|altura|-|integer|20-65535 (padrão: 200)|Altura do gráfico, em pixels. Usado para visualização e para gráficos de pizza/explodidos.|<|yaxismin|-|double|Padrão: 0|Valor mínimo do eixo Y.

Usado se 'ymin_type_1' for FIXED.|<|yaxismax|-|double|Padrão: 0|Valor máximo do eixo Y.

Usado se 'ymax_type_1' for FIXED.|<|show_work_period|-|string|0 - NÃO
1 - SIM (padrão)|Realçar horas de folga.

Usado por gráficos normais e empilhados.|<|show_triggers|-|string|0 - NO
1 - YES (padrão)|Exibe valores de trigger simples como uma linha.

Usado por gráficos normais e empilhados.|<|tipo|-|string|0 - NORMAL (padrão)
1 - EMPILHADO
2 - TORTA
3 - EXPLODIDO|Tipo de gráfico.|<|show_legend|-|string|0 - NÃO
1 - SIM (padrão)|Exibe legenda do gráfico.|<|show_3d|-|string|0 - NO (padrão)
1 - YES|Ativar estilo 3D.

Usado por pizza e gráficos de pizza explodidos.|<|percent_left|-|double|Default:0|Mostra a linha de percentil para o eixo esquerdo.

Usado apenas para gráficos normais.|<|percent_right|-|double|Default:0|Mostra a linha de percentil para o eixo direito.

Usado apenas para gráficos normais.|<|ymin_type_1|-|string|0 - CALCULATED (padrão)
1 - FIXED
2 - ITEM|Valor mínimo do eixo Y.

Usado por normal e gráficos empilhados.|<|ymax_type_1|-|string|0 - CALCULATED (padrão)
1 - FIXED
2 - ITEM|Valor máximo do eixo Y.

Usado por normal e gráficos empilhados.|ymin_item_1|<|<|<|Detalhes do item individual.

Obrigatório se 'ymin_type_1' for ITEM.|<|host|x|string|<|Anfitrião do item.|<|chave|x|string|<|chave do item.|ymax_item_1|<|<|<|Detalhes do item individual.

Obrigatório se 'ymax_type_1' for ITEM.|<|host|x|string|<|Anfitrião do item.|<|chave|x|string|<|chave do item.|graph_items|<x|<|<|Elemento raiz para itens de gráfico.|<|sortorder|-|integer|<|Ordem de sorteio. O valor menor é sorteado primeiro. Pode ser usado para desenhar linhas ou regiões atrás (ou na frente) de outra.|<|drawtype|-|string|0 - SINGLE_LINE (padrão)
1 - FILLED_REGION
2 - BOLD_LINE
3 - DOTTED_LINE
4 - TRAÇADO_LINE
5 - GRADIENT_LINE|Estilo de desenho do item gráfico.

Usado apenas por gráficos normais.|<|color|-|string|<|Cor do elemento (6 símbolos, hex).|<|yaxiside|-|string|0 - LEFT (padrão)
1 - RIGHT|Lado do gráfico onde a escala Y do item do gráfico será desenhada.

Usado por normal e empilhado gráficos.|<|calc_fncl|-|string|1 - MIN
2 - AVG (padrão)
4 - MAX
7 - ALL (mínimo, médio e máximo; usado apenas por gráficos simples)
9 - LAST (usado apenas por gráficos de pizza e gráficos de pizza explodidos)|Dados a serem desenhados se existir mais de um valor para um item.|<|type|-|string|0 - SIMPLE (padrão)
2 - GRAPH_SUM (o valor do item representa a pizza inteira; usado apenas por gráficos de pizza e explodidos)|Tipo de item de gráfico. |item|<x|<|<|item individual.|<|host|x|string|<|Anfitrião do item.|<|chave|x|string|<|chave do item.|

Tags de cenário da web de modelo

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|-----|-----|-----|
|httptests|<|<|<|Elemento raiz para cenários da web.|<|uuid|x|string|<|Identificador exclusivo para este cenário da web.|<|name|x|string|<|Nome do cenário da Web.|<|delay|-|string|Padrão: 1m|Frequência de execução do cenário web. Segundos, unidade de tempo com sufixo ou macro de usuário.|<|attempts|-|integer|1-10 (padrão: 1)|O número de tentativas para executar as etapas do cenário web.|<|agente|-|string|Padrão: Zabbix|Agente cliente. O Zabbix fingirá ser o navegador selecionado. Isso é útil quando um site retorna conteúdo diferente para navegadores diferentes.|<|http_proxy|-|string|<|Especifique um proxy HTTP a ser usado, usando o formato: http://[username[:password]@]proxy.example.com[:port] |variables|<|<|<|Elemento raiz para variáveis de nível de cenário (macros) que podem ser usadas em etapas de cenário.|<|nome|x|texto|<|Nome da variável.|<|valor|x|texto|<|Valor da variável.|headers|<|<|<|Elemento raiz para cabeçalhos HTTP que serão enviados ao realizar uma solicitação. Os cabeçalhos devem ser listados usando a mesma sintaxe que apareceriam no protocolo HTTP.|<|nome|x|texto|<|Nome do cabeçalho.|<|valor|x|texto|<|Valor do cabeçalho.|<|status|-|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do cenário da Web.|<|autenticação|-|string|0 - NONE (padrão)
1 - BA-

SIC
2 - NTLM|Método de autenticação.| |<|http_user|-|string|<|Nome de usuário usado para autenticação básica, HTTP ou NTLM.| |<|http_password|-|string|<|Senha usada para autenticação básica, HTTP ou NTLM.| |<|verify_peer|-|string|0 - NO (padrão)
1 - YES|Verifique o certificado SSL do servidor web.| |<|verify_host|-|string|0 - NO (padrão)
1 - YES|Verifique se o campo Common Name ou o campo Subject Alternate Name do certificado do servidor web corresponde.| |<|ssl_cert_file|-|string|<|Nome do arquivo de certificado SSL usado para autenticação do cliente (deve estar no formato PEM).| |<|ssl_key_file|-|string|<|Nome do arquivo de chave privada SSL usado para autenticação do cliente (deve estar no formato PEM).| |<|ssl_key_password|-|string|<|Senha do arquivo de chave privada SSL.| |steps|<|x|<|<|Elemento raiz para etapas do cenário da web.| |<|name|x|string|<|Nome da etapa do cenário da Web.| |<|url|x|string|<|URL para monitoramento.| |query_fields|<|<|<|Elemento raiz para campos de consulta - uma matriz de campos HTTP que serão adicionados à URL ao realizar uma solicitação.| |<|nome|x|string|<|Nome do campo de consulta.| |<|valor|-|string|<|Valor do campo de consulta.| |posts|<|<|<|Variáveis HTTP POST como uma string (dados brutos de postagem) ou como uma matriz de campos HTTP (dados de campo de formulário).| |<|nome|x|string|<|Nome do campo de postagem.| |<|valor|x|string|<|Valor do campo de postagem.| |variables|<|<|<|Elemento raiz das variáveis de nível de etapa (macros) que devem ser aplicadas após esta etapa.

Se o valor da variável tiver um prefixo 'regex:', então seu value é extraído dos dados retornados por esta etapa de acordo com o padrão de expressão regular seguindo o prefixo 'regex:' |<|nome|x|string|<|Nome da variável.| |<|valor|x|string|<|Valor da variável.| |headers|<|<|<|Elemento raiz para cabeçalhos HTTP que serão enviados ao realizar uma solicitação. Os cabeçalhos devem ser listados usando a mesma sintaxe que apareceriam no protocolo HTTP.| |<|nome|x|string|<|Nome do cabeçalho.| |<|valor|x|string|<|Valor do cabeçalho.| |<|follow_redirects|-|string|0 - NÃO
1 - SIM (padrão)|Siga redirecionamentos HTTP.| |<|retrieve_model|-|string|0 - BODY (padrão)
1 - HEADERS
2 - BOTH|Modo de recuperação de resposta HTTP.| |<|timeout|-|string|Padrão: 15s|Tempo limite de execução da etapa. Segundos, unidade de tempo com sufixo ou macro de usuário.| |<|required|-|string|<|Texto que deve estar presente na resposta. Ignorado se estiver vazio.| |<|status_codes|-|string|<|Uma lista delimitada por vírgulas de códigos de status HTTP aceitos. Ignorado se estiver vazio. Por exemplo: 200-201,210-299| |tags|<|<|<|Elemento raiz para tags de cenário da web.| |<|tag|x|string|<|Nome da tag.| |<|valor|-|string|<|Valor da tag.|

Tags de painel de modelo

Elemento	Propriedade do elemento	Requerido	Tipo	Intervalo ¹	Descrição	
painéis < < <	Elemento raiz para painéis de modelo.	< uuid x string	Identificador exclusivo para este painel.			
< name x string	Nome do painel do modelo.	< exibir período - integer	Exibir período das páginas do painel.			
< auto_start - string	0 - não 1 - sim	Início automático da apresentação de slides.	pages < < <	Elemento raiz para páginas de painel de modelo.		
< nome - string	Nome da página.	< período de exibição - inteiro	Período de exibição da página.	< sortorder - integer	Ordem de classificação das páginas.	
< widgets < < <	Elemento raiz para widgets de painel de modelo.	< tipo x string	Tipo de widget.	< nome - string	Nome do widget.	
< x - integer	0-23	Posição horizontal do lado esquerdo do painel do modelo.	< y - integer	0-62	Posição vertical da parte superior do painel do modelo.	
< largura - inteiro	1-24	Largura do widget.	< altura - inteiro	2-32	Altura do widget.	
< hide_header - string	0 - não 1 - sim	Ocultar cabeçalho do widget.	fields < < <	Elemento raiz para os campos do widget do painel de modelo.		
< type x string	0 - INTEGER 1 - STRING 3 - HOST 4 - ITEM 5 - ITEM_PROTOTYPE 6 - GRAPH 7 - GRAPH_PROTOTYPE	Tipo de campo de widget.	< nome x string	Nome do campo do widget.	< value x mixed	Valor do campo do widget, dependendo do tipo de campo.

Notas de rodapé

¹ Para valores de string, apenas a string será exportada (por exemplo, "ZABBIX_ACTIVE") sem a numeração utilizada nesta tabela. Os números para valores de intervalo (correspondentes aos valores da API) nesta tabela é usado apenas para encomenda.

3 Hosts

Visão geral

Os hosts são **exportados** com muitos objetos e relações objetais.

A exportação do host contém:

- grupos de hosts vinculados
- dados do host
- link de modelo
- ligação do grupo de hosts
- interfaces de host
- itens diretamente vinculados
- gatilhos diretamente ligados
- gráficos diretamente ligados
- regras de descoberta diretamente vinculadas com todos os protótipos
- cenários da web diretamente vinculados

- macros de host
- dados de inventário do host
- mapas de valor

Exportando

Para exportar hosts, faça o seguinte:

- Vá para: Configuração → Hosts
- Marque as caixas de seleção dos hosts para exportar
- Clique em Exportar abaixo da lista

≡ Hosts

<input type="checkbox"/>	Name ▲	Items	Triggers	Graphs	Discovery	Web
<input checked="" type="checkbox"/>	Server1	Items	Triggers	Graphs	Discovery	Web

1 selected

Enable
Disable
Export

YAML
XML
JSON

Mass update
Delete

Dependendo do formato selecionado, os hosts são exportados para um arquivo local com um nome padrão:

- zabbix_export_hosts.yaml - na exportação YAML (opção padrão para exportar)
- zabbix_export_hosts.xml - na exportação XML
- zabbix_export_hosts.json - na exportação JSON

Importando

Para importar hosts, faça o seguinte:

- Vá para: Configuração → Hosts
- Clique em Importar à direita
- Selecione o arquivo de importação
- Marque as opções necessárias nas regras de importação
- Clique em Importar

Import

* Import file

Choose file

zbx_export_hosts.yaml

Rules	Update existing	Create new	Delete missing
Groups	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Hosts	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Value mappings	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Template linkage		<input checked="" type="checkbox"/>	<input type="checkbox"/>
Items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Discovery rules	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Graphs	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Web scenarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Import

Cancel

Uma mensagem de sucesso ou falha da importação será exibida na parte dianteira.

Regras de importação:

Regra	Descrição
Atualizar existente	Os elementos existentes serão atualizados com os dados retirados do arquivo de importação. Caso contrário, eles não serão atualizados.
Criar novo	A importação adicionará novos elementos usando dados do arquivo de importação. Caso contrário, não irá adicioná-los.
Delete missing	A importação removerá elementos existentes não presentes no arquivo de importação. Caso contrário, ele não os removerá. Se Excluir ausente estiver marcado para vinculação de modelo, a vinculação de modelo existente não presente no arquivo de importação será removida do host junto com todas as entidades herdadas dos modelos potencialmente desvinculados (itens, acionadores , etc).

Formato de exportação

Formato de exportação em YAML:

```
zabbix_export:
  versão: '6.0'
  data: '2021-09-28T12:20:07Z'
  grupos:
    -
      uuid: f2481361f99448eea617b7b1d4765566
      nome: 'Anfitriões descobertos'
    -
      uuid: 6f6799aa69e844b4b3918f779f2abf08
      nome: 'servidores Zabbix'
  anfitriões:
    -
      host: 'servidor Zabbix 1'
```

```

nome: 'Servidor principal do Zabbix'
modelos:
-
  nome: 'Linux pelo agente Zabbix'
-
  nome: 'Saúde do servidor Zabbix'
grupos:
-
  nome: 'Anfitriões descobertos'
-
  nome: 'servidores Zabbix'
interfaces:
-
  ip: 192.168.1.1
  interface_ref: if1
Unid:
-
  nome: 'armadilha Zabbix'
  tipo: TRAP
  chave: armadilha
  atraso: '0'
  história: 1w
  pré-processando:
  -
    tipo: MULTIPLICADOR
    parâmetros:
      - '8'
  Tag:
  -
    etiqueta: aplicativo
    valor: 'servidor Zabbix'
  gatilhos:
  -
    expressão: 'last(/servidor Zabbix 1/trap)=0'
    nome: 'Último valor é zero'
    prioridade: AVISO
    Tag:
    -
      etiqueta: processo
      valor: 'Teste interno'
  Tag:
  -
    etiqueta: processo
    valor: Zabbix
macros:
-
  macro: '{$HOST.MACRO}'
  valor: '123'
-
  macro: '{$PASSWORD1}'
  tipo: SECRET_TEXT
inventário:
  digite: 'servidor Zabbix'
  nome: yyyyyy-HP-Pro-3010-Small-Form-Factor-PC
  os: 'Linux yyyyyy-HP-Pro-3010-Small-Form-Factor-PC 4.4.0-165-generic #193-Ubuntu SMP Tue Set 17 17
Inventory_mode: AUTOMÁTICO
gráficos:
-
  nome: 'servidor de utilização da CPU'
  show_work_period: 'NÃO'
  show_triggers: 'NÃO'
  graph_items:

```

```

-
tipo de desenho: FILLED_REGION
cor: FF5555
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,roubar]'
-
ordem de classificação: '1'
tipo de desenho: FILLED_REGION
cor: 55FF55
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,softirq]'
-
ordem de classificação: '2'
tipo de desenho: FILLED_REGION
cor: '009999'
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,interromper]'
-
ordem de classificação: '3'
tipo de desenho: FILLED_REGION
cor: '990099'
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,nice]'
-
ordem de classificação: '4'
tipo de desenho: FILLED_REGION
cor: '999900'
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,iowait]'
-
ordem de classificação: '5'
tipo de desenho: FILLED_REGION
cor: '990000'
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,system]'
-
ordem de classificação: '6'
tipo de desenho: FILLED_REGION
cor: '000099'
calc_fnc: MIN
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,usuário]'
-
ordem de classificação: '7'
tipo de desenho: FILLED_REGION
cor: '009900'
item:
  host: 'servidor Zabbix 1'
  chave: 'system.cpu.util[,idle]'

```

Tags de elemento

Os valores de tag de elemento são explicados na tabela abaixo.

Tags de host

Elemento	Propriedade do elemento	Requerido	Tipo	Intervalo ¹	Descrição

|groups|<|x|<|Elemento raiz para grupos de hosts.|<|name|x|string|<|Nome do grupo de hosts.| |hosts|<|<|Elemento raiz para hosts.|<|host|x|string|<|Nome de host exclusivo.|<|name|-|string|<|Nome do host visível.|<|descricao|-|text|<|Descrição do host.|<|status|-|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do host.|<|ipmi_authtype|-|string|1 - DEFAULT (padrão)
0 - NONE
1 - MD2
2 - MD5
4 - STRAIGHT
5 - OEM
6 - RMCP_PLUS|tipo de autenticação de sessão IPMI.|<|ipmi_privilege|-|string|1 - CALLBACK
2 - USER (padrão)
3 - OPERATOR
4 - ADMIN
5 - OEM|IPMI nível de privilégio de sessão. |<|ipmi_username|-|string|<|Nome de usuário para verificações de IPMI.|<|ipmi_password|-|string|<|Senha para verificações de IPMI.| |proxy|<|<|<|Proxy.|<|name|x|string|<|Nome do proxy (se houver) que monitora o host.| |templates|<|<|<|Elemento raiz para templates vinculados.|<|name|x|string|<|Nome do modelo.| |interfaces|<|<|<|Elemento raiz para interfaces de host.|<|default|-|string|0 - NO
1 - YES (default)|Se esta é a interface principal do host.
Só pode haver uma interface primária de um tipo em um host. |<|tipo|-|string|1 - ZABBIX (padrão)
2 - SNMP
3 - IPMI
4 - JMX|Tipo de interface.|<|useip|-|string|0 - NÃO
1 - SIM (padrão)|Se usar IP como interface para conexão com o host (se não, DNS será usado).|<|ip|-|string|<|endereço IP, pode ser IPv4 ou IPv6.

Obrigatório se a conexão for feita via IP.|<|dns|-|string|<|Nome DNS.

Obrigatório se a conexão for feita via DNS.|<|port|-|string|<|Número da porta. Suporta macros de usuário.|<|interface_ref|x|string|Formato: if<N>|Nome de referência de interface a ser usado em itens.| |details|<|<|<|Elemento raiz para detalhes da interface.|<|versão|-|string|1 - SNMPV1
2 - SNMP_V2C (padrão)
3 - SNMP_V3|Use esta versão SNMP.|<|comunidade|-|string|<|Comunidade SNMP.

Requerido para itens SNMPv1 e SNMPv2.|<|contextname|-|string|<|Nome do contexto SNMPv3.

Usado apenas por itens SNMPv3.|<|securityname|-|string|<|Nome de segurança SNMPv3.

Usado apenas por itens SNMPv3.|<|securitylevel|-|string|0 - NOAUTHNOPRIV (padrão)
1 - AUTHNOPRIV
2 - AUTHPRIV|Nível de segurança SNMPv3.

Usado apenas por itens SNMPv3.|<|authprotocol|-|string|0 - MD5 (padrão)
1 - SHA1
2 - SHA224
3 - SHA256
4 - SHA384
5 - SHA512|SNMPv3 protocolo de autenticação.

Usado apenas por itens SNMPv3.|<|authpassphrase|-|string|<|Senha de autenticação SNMPv3.

Usado apenas por itens SNMPv3.|<|privprotocol|-|string|0 - DES (padrão)
1 - AES128
2 - AES192
3 - AES256
4 - AES192C
5 - AES256C|SNMPv3 protocolo de privacidade.

Usado apenas por itens SNMPv3.|<|privpassphrase|-|string|<|Senha de privacidade SNMPv3.

Usado apenas por itens SNMPv3.|<|bulk|-|string|0 - NÃO
1 - SIM (padrão)|Use solicitações em massa para SNMP.| |items|<|<|<|Elemento raiz para itens.|<|Para valores de tag de elemento de item, consulte tags de host **item**.|<|<|<|<|tags|<|<|<|Elemento raiz para tags de host.|<|tag|x|string|<|Nome da tag.|<|valor|-|string|<|Valor da tag.| |macros|<|<|<|Elemento raiz para macros.|<|macro|x|<|<|Nome da macro do usuário.|<|type|-|string|0 - TEXT (padrão)
1 - SECRET_TEXT
2 - VAULT|Tipo da macro.|<|valor|-|string|<|Valor da macro do usuário.|<|descricao|-|string|<|Descrição da macro do usuário.| |inventário|<|<|<|Elemento raiz para inventário do host.|<|<|inventário_property>|<|<|<|Propriedade de inventário individual.

Todas as propriedades de inventário disponíveis estão listadas nas respectivas tags, por exemplo, <type>, <name>, <os> (veja o exemplo acima).| |inventory_mode|<|-|string|1 - DESATIVADO
0 - MANUAL (padrão)
1 - AUTOMÁTICO|Modo de inventário.| |valuemaps|<|<|<|Elemento raiz para mapas de valor do host.|<|name|x|string|<|Nome do mapa de valores.|<|mapping|<|<|<|Elemento raiz para mapeamentos.|<|valor|x|string|<|Valor de um mapeamento.|<|newvalue|x|string|<|Novo valor de um mapeamento.|

Tags de itens do host

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|-----|-----|-----|
| |items|<|<|<|Elemento raiz para itens.|<|name|x|string|<|Nome do item.|<|type|-|string|0 - ZABBIX_PASSIVE (padrão)
2 - TRAP
3 - SIMPLE
5 - INTERNO
7 - ZABBIX_ACTIVE
10 - EXTERNO
11 - ODBC
12 - IPMI
13 - SSH
14 - TELNET
15 - CALCULADO
16 - JMX
17 - SNMP_TRAP
18 - DEPENDENT
19 - HTTP_AGENT
20 - SNMP_AGENT
21 - ITEM_TYPE_SCRIPT|Tipo de item.|<|snmp_oid|-|string|<|ID do objeto SNMP.

Requerido por itens SNMP.|<|chave|x|string|<|chave do item.|<|delay|-|string|Padrão: 1m|Intervalo de atualização do item.

Observe que delay será sempre '0' para itens trapper.

Aceita segundos ou uma unidade de tempo com sufixo (30s, 1m, 2h, 1d).

Opcionalmente, um ou mais **intervalos personalizados** podem ser especificados como intervalos flexíveis ou agendamento.

Vários intervalos são separados por um ponto e vírgula.

As macros do usuário podem ser usadas. Uma única macro deve preencher todo o campo. Múltiplas macros em um campo ou macros misturadas com texto não são suportadas.

Intervalos flexíveis podem ser escritos como duas macros separadas por uma barra (por exemplo, {\$FLEX_INTERVAL}/{FLEX_PERIOD}).|<|history|-|string|Padrão: 90d|Uma unidade de tempo de quanto tempo os dados do histórico devem ser armazenados. Unidade de tempo com sufixo, macro de usuário ou macro LLD.|<|tendências|-|string|Padrão: 365d|Uma unidade de tempo de quanto tempo os dados de tendências devem ser armazenados. Unidade de tempo com sufixo, macro de usuário ou macro LLD.|<|status|-|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do item.|<|value_type|-|string|0 - FLOAT
1 - CHAR
2 - LOG
3 - UNSIGNED (padrão)
4 - TEXT|Tipo de valor recebido.|<|allowed_hosts|-|string|<|Lista de endereços IP (delimitados por vírgula) de hosts com permissão para enviar dados para o item.

Usado por itens do agente trapper e HTTP.|<|unidades|-|string|<|Unidades de valores retornados (bps, B, etc).|<|params|-|text|<|Parâmetros adicionais dependendo do tipo do item:
- script executado para itens de Script, SSH e Telnet;
- consulta SQL para itens de monitor de banco de dados;
- fórmula para itens calculados.|<|ipmi_sensor|-|string|<|Sensor IPMI.

Usado apenas por itens IPMI.|<|authtype|-|string|Tipo de autenticação para itens do agente SSH:
0 - SENHA (padrão)
1 - PUBLIC_KEY

Tipo de autenticação para itens do agente HTTP:
0 - NONE (padrão)
1 - BASIC
2 - NTLM|Tipo de autenticação.

Usado apenas por itens de agente SSH e HTTP.|<|username|-|string|<|Nome de usuário para autenticação.

Usado por verificação simples, SSH, Telnet, monitor de banco de dados, itens de agente JMX e HTTP.

Requerido por SSH e Telnet itens.

Quando usado pelo agente JMX, a senha também deve ser especificada junto com o nome de usuário ou ambas as propriedades devem ser deixadas em

branco. |<|password|-|string|<|Senha para autenticação.
Usado por verificação simples, SSH, Telnet, monitor de banco de dados, itens de agente JMX e HTTP.

Quando usado pelo agente JMX, o nome de usuário também deve ser especificado junto com a senha ou ambas as propriedades devem ser deixadas em branco. |<|publickey|-|string|<|Nome do arquivo de chave pública.

Obrigatório para itens do agente SSH. |<|privatekey|-|string|<|Nome do arquivo de chave privada.

Obrigatório para itens do agente SSH. |<|descrição|-|text|<|Descrição do item. |<|inventory_link|-|string|0 - NONE

Nome do campo de inventário do host em letras maiúsculas. Por exemplo:
4 - ALIAS
6 - OS_FULL
14 - HARDWARE
etc. |<|Campo de inventário do host que é preenchido pelo item.

Consulte o [host página de inventário](#) para obter uma lista de campos de inventário de host compatíveis e seus IDs. |<|logtimefmt|-|string|<|Formato da hora nas entradas de log.
Usado apenas por itens de log. |<|interface_ref|-|string|<|Formato: if<N>|Referência à interface do host. |<|jmx_endpoint|-|string|<|JMX endpoint.

Usado apenas por itens do agente JMX. |<|url|-|string|<|String de URL.

Requerido apenas para itens de agente HTTP. |<|allow_traps|-|string|0 - NO (padrão)
1 - YES|Permitir preencher o valor como em um item de trapper.

Usado apenas por itens de agente HTTP. |<|follow_redirects|-|string|0 - NÃO
1 - SIM (padrão)|Seguir redirecionamentos de resposta HTTP ao agrupar dados.

Usado apenas por itens de agente HTTP. |headers|<|<|<|Elemento raiz para cabeçalhos de solicitação HTTP(S), em que o nome do cabeçalho é usado como chave e o valor do cabeçalho como valor.
Usado apenas por itens de agente HTTP. |<|nome|x|string|<|Nome do cabeçalho. |<|valor|x|string|<|Valor do cabeçalho. |<|http_proxy|-|string|<|String de conexão do proxy HTTP(S).

Usado apenas por itens do agente HTTP. |<|output_format|-|string|0 - RAW (padrão)
1 - JSON|Como processar a resposta.

Usado apenas por itens do agente HTTP. |<|post_type|-|string|0 - RAW (padrão)
2 - JSON
3 - XML|Tipo de corpo de dados da postagem.

Usado apenas pelo agente HTTP item

Hospedar tags de regra de descoberta de baixo nível

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|----|-----| |discovery_rules|<|<|<|Elemento raiz para regras de descoberta de baixo nível. |<|Para a maioria dos valores de tag de elemento, veja valores de tag de elemento para um item regular. Somente as tags específicas para regras de descoberta de baixo nível são descritas abaixo.<|<|<|<|<|type|-|string|0 - ZABBIX_PASSIVE (padrão)
2 - TRAP
3 - SIMPLE
5 - INTERNO
7 - ZABBIX_ACTIVE
10 - EXTERNO
11 - ODBC
12 - IPMI
13 - SSH
14 - TELNET
16 - JMX
18 - DEPENDENTE
19 - HTTP_AGENT
20 - SNMP_AGENT|Tipo de item. |<|vida|-|string|Padrão: 30d|Período de tempo após o qual os itens que não são mais descobertos serão excluídos. Segundos, unidade de tempo com sufixo ou macro de usuário. |filtro|<|<|<|<|Filtro individual. |<|evaltype|-|string|0 - AND_OR (padrão)
1 - AND
2 - OU
3 - FORMULA|Lógica a ser usada para verificar o filtro de regra de descoberta de baixo nível condições. |<|formula|-|string|<|Fórmula de cálculo personalizada para condições de filtro. |conditions|<|<|<|Elemento raiz para condições de filtro. |<|macro|x|string|<|Nome da macro de descoberta de baixo nível. |<|valor|-|string|<|Valor do filtro: expressão regular ou expressão regular global. |<|operador|-|string|8 - MATCHES_REGEX (padrão)
9 - NOT_MATCHES_REGEX|Operador de condição. |<|formulaid|x|character|<|ID exclusivo arbitrário que é usado para fazer referência a uma condição da expressão personalizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente. |lld_macro_paths|<|<|<|Elemento raiz para caminhos de macro LLD. |<|lld_macro|x|string|<|Nome da macro de descoberta de baixo nível. |<|path|x|string|<|Seletor para o valor que será atribuído à macro correspondente. |pré-processamento|<|<|<|pré-processamento do valor da regra LLD. |step|<|<|<|Etapa de pré-processamento do valor da regra LLD individual. |<|Para a maioria dos valores de tag de elemento, consulte valores de tag de elemento para um pré-processamento de valor de item de host. Somente as tags que são específicas para o pré-processamento de valor de descoberta de baixo nível são descritas abaixo.<|<|<|<|<|type|x|string|5 - REGEX
11 - XMLPATH
12 - JSONPATH
15 - NOT_MATCHES_REGEX
16 - CHECK_JSON_ERROR
17 - CHECK_XML_ERROR
20 - DISCARD_UNCHANGED_HEARTBEAT
21 - JAVASCRIPT
23 - PROMETHEUS_TO_JSON
24 - CSV_TO_JSON
25 - STR_REPLACE
27 - XML_TO_JSON|Tipo da etapa de pré-processamento do valor do item. |trigger_prototypes|<|<|<|Elemento raiz para protótipos de trigger. |<|Para valores de tag de elemento protótipo de acionador, consulte tags regulares [host trigger](#).<|<|<|<|graph_prototypes|<|<|<|Elemento raiz para protótipos de gráfico. |<|Para valores de tag de elemento de protótipo de gráfico, consulte tags regulares [host graph](#).<|<|<|<|host_prototypes|<|<|<|Elemento raiz para protótipos de host. |<|Para valores de tag de elemento de protótipo de host, consulte tags regulares [host](#).<|<|<|<|item_prototypes|<|<|<|Elemento raiz para protótipos de itens. |<|Para valores de tag de elemento de protótipo de item, consulte tags regulares [host item](#).<|<|<|<|master_item|<|<|<|Dados de protótipo de item/protótipo de item de protótipo de item individual. |<|key|x|string|<|Valor da chave do protótipo do item mestre/protótipo do item dependente.

Obrigatório para um item dependente. |

Tags de gatilho do host

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|----|-----| |triggers|<|<|<|Elemento raiz para triggers. |<|expressão|x|string|<|Expressão de gatilho. |<|recovery_mode|-|string|0 - EXPRESSION (padrão)
1 - RECOVERY_EXPRESSION
2 - NONE|Base para geração de eventos OK. |<|recuperação_expressão|-|string|<|Expressão de recuperação de gatilho. |<|nome|x|string|<|Nome do gatilho. |<|correlation_mode|-|string|0 - DISABLED (padrão)
1 - TAG_VALUE|Modo de correlação (sem correlação de evento ou correlação de evento por tag). |<|correlation_tag|-|string|<|O nome do tag a ser usado para correlação de eventos. |<|url|-|string|<|URL associado ao gatilho. |<|status|-|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do gatilho. |<|prioridade|-|string|0 - NOT_CLASSIFIED (padrão)
1 - INFO
2 - AVISO
3 - MÉDIA
4 - ALTA
5 - DESASTRE |Gravidade do gatilho. |<|descrição|-|text|<|Descrição do gatilho. |<|type|-|string|0 - SINGLE (padrão)
1 - MULTIPLE|Tipo de geração de evento (evento de problema único ou vários eventos de problema). |<|manual_close|-|string|0 - NÃO

(padrão)
1 - SIM|Fechamento manual de eventos de problema.| |dependencies|<|<|<|Elemento raiz para dependências.| |<|nome|x|string|<|Nome do gatilho de dependência.| |<|expression|x|string|<|Expressão do gatilho de dependência.| |<|recuperação_expression|<|string|<|Expressão de recuperação do gatilho de dependência.| |tags|<|<|<|Elemento raiz para tags de evento.| |<|tag|x|string|<|Nome da tag.| |<|valor|<|string|<|Valor da tag.|

Tags do gráfico do host

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|----|-----|-----|
|gráficos|<|<|<|Elemento raiz para gráficos.| |<|nome|x|string|<|Nome do gráfico.| |<|width|<|integer|20-65535 (padrão: 900)|Largura do gráfico, em pixels. Usado para visualização e para gráficos de pizza/explodidos.| |<|altura|<|inteiro|20-65535 (padrão: 200)|Altura do gráfico, em pixels. Usado para visualização e para gráficos de pizza/explodidos.| |<|yaxismin|<|double|Padrão: 0|Valor mínimo do eixo Y.

Usado se 'ymin_type_1' for FIXED.| |<|yaxismax|<|double|Padrão: 0|Valor máximo do eixo Y.

Usado se 'ymax_type_1' for FIXED.| |<|show_work_period|<|string|0 - NÃO
1 - SIM (padrão)|Realçar horas de folga.

Usado por gráficos normais e empilhados.| |<|show_triggers|<|string|0 - NO
1 - YES (padrão)|Exibe valores de trigger simples como uma linha.

Usado por gráficos normais e empilhados.| |<|tipo|<|string|0 - NORMAL (padrão)
1 - EMPILHADO
2 - TORTA
3 - EXPLODIDO|Tipo de gráfico.| |<|show_legend|<|string|0 - NÃO
1 - SIM (padrão)|Exibe legenda do gráfico.| |<|show_3d|<|string|0 - NO (padrão)
1 - YES|Ativar estilo 3D.

Usado por pizza e gráficos de pizza explodidos.| |<|percent_left|<|double|Default:0|Mostra a linha de percentil para o eixo esquerdo.

Usado apenas para gráficos normais.| |<|percent_right|<|double|Default:0|Mostra a linha de percentil para o eixo direito.

Usado apenas para gráficos normais.| |<|ymin_type_1|<|string|0 - CALCULATED (padrão)
1 - FIXED
2 - ITEM|Valor mínimo do eixo Y.

Usado por normal e gráficos empilhados.| |<|ymax_type_1|<|string|0 - CALCULATED (padrão)
1 - FIXED
2 - ITEM|Valor máximo do eixo Y.

Usado por normal e gráficos empilhados.| |ymin_item_1|<|<|<|Detalhes do item individual.

Obrigatório se 'ymin_type_1' for ITEM.| |<|host|x|string|<|Anfitrião do item.| |<|chave|x|string|<|chave do item.| |ymax_item_1|<|<|<|Detalhes do item individual.

Obrigatório se 'ymax_type_1' for ITEM.| |<|host|x|string|<|Anfitrião do item.| |<|chave|x|string|<|chave do item.| |graph_items|<|x|<|<|Elemento raiz para itens de gráfico.| |<|sortorder|<|integer|<|Ordem de sorteio. O valor menor é sorteado primeiro. Pode ser usado para desenhar linhas ou regiões atrás (ou na frente) de outra.| |<|drawtype|<|string|0 - SINGLE_LINE (padrão)
1 - FILLED_REGION
2 - BOLD_LINE
3 - DOTTED_LINE
4 - TRAÇADO_LINE
5 - GRADIENT_LINE|Estilo de desenho do item gráfico.

Usado apenas por gráficos normais.| |<|color|<|string|<|Cor do elemento (6 símbolos, hex).| |<|yaxiside|<|string|0 - LEFT (padrão)
1 - RIGHT|Lado do gráfico onde a escala Y do item do gráfico será desenhada.

Usado por normal e empilhado gráficos.| |<|calc_fnc|<|string|1 - MIN
2 - AVG (padrão)
4 - MAX
7 - ALL (mínimo, médio e máximo; usado apenas por gráficos simples)
9 - LAST (usado apenas por gráficos de pizza e gráficos de pizza explodidos)|Dados a serem desenhados se existir mais de um valor para um item.| |<|type|<|string|0 - SIMPLE (padrão)
2 - GRAPH_SUM (o valor do item representa a pizza inteira; usado apenas por gráficos de pizza e explodidos)|Tipo de item de gráfico. | |item|<|x|<|<|item individual.| |<|host|x|string|<|Anfitrião do item.| |<|chave|x|string|<|chave do item.|

Hospedar tags de cenário da web

|Elemento|Propriedade do elemento|Requerido|Tipo|Intervalo¹|Descrição| |-----|-----|-----|----|-----|-----|
--| |httptests|<|<|<|Elemento raiz para cenários da web.| |<|name|x|string|<|Nome do cenário da Web.| |<|delay|<|string|Padrão: 1m|Frequência de execução do cenário web. Segundos, unidade de tempo com sufixo ou macro de usuário.| |<|attempts|<|integer|1-10 (padrão: 1)|O número de tentativas para executar as etapas do cenário web.| |<|agent|<|string|Padrão: Zabbix|Agente cliente. O Zabbix fingirá ser o navegador selecionado. Isso é útil quando um site retorna conteúdo diferente para navegadores diferentes.| |<|http_proxy|<|string|<|Especifique um proxy HTTP a ser usado, usando o formato: http://[username[:password]@]proxy.example.com[:port] | |variables|<|<|<|Elemento raiz para variáveis de nível de cenário (macros) que podem ser usadas em etapas de cenário.| |<|nome|x|texto|<|Nome da variável.| |<|valor|x|texto|<|Valor da variável.| |headers|<|<|<|Elemento raiz para cabeçalhos HTTP que serão enviados ao realizar uma solicitação. Os cabeçalhos devem ser listados usando a mesma sintaxe que aparecerem no protocolo HTTP.| |<|nome|x|texto|<|Nome do cabeçalho.| |<|valor|x|texto|<|Valor do cabeçalho.| |<|status|<|string|0 - ATIVADO (padrão)
1 - DESATIVADO|Status do cenário da Web.| |<|autenticação|<|string|0 - NONE (padrão)
1 - BASIC
2 - NTLM|Método de autenticação.| |<|http_user|<|string|<|Nome de usuário usado para autenticação básica, HTTP ou NTLM.| |<|http_password|<|string|<|Senha usada para autenticação básica, HTTP ou NTLM.| |<|verify_peer|<|string|0 - NO (padrão)
1 - YES|Verifique o certificado SSL do servidor web.| |<|verify_host|<|string|0 - NO (padrão)
1 - YES|Verifique se o campo Common Name ou o campo Subject Alternate Name do certificado do servidor web corresponde.| |<|ssl_cert_file|<|string|<|Nome do arquivo de certificado SSL usado para autenticação do cliente (deve estar no formato PEM).| |<|ssl_key_file|<|string|<|Nome do arquivo de chave privada SSL usado para autenticação do cliente (deve estar no formato PEM).| |<|ssl_key_password|<|string|<|Senha do arquivo de chave privada SSL.| |steps|<|x|<|<|Elemento raiz para etapas do cenário da web.| |<|nome|x|string|<|Nome da etapa do cenário da Web.| |<|url|x|string|<|URL para monitoramento.| |query_fields|<|<|<|Elemento raiz para campos de consulta - uma matriz de campos HTTP que serão adicionados à URL ao realizar uma solicitação.| |<|nome|x|string|<|Nome do campo de consulta.| |<|valor|<|string|<|Valor do campo de consulta.| |posts|<|<|<|Variáveis HTTP POST como uma string (dados brutos de postagem) ou como uma matriz de campos HTTP (dados de campo de formulário).| |<|nome|x|string|<|Nome do campo de postagem.| |<|valor|x|string|<|Valor do campo de postagem.| |variables|<|<|<|Elemento raiz das variáveis de nível de etapa (macros) que devem ser aplicadas após esta etapa.

Se o valor da variável tiver um prefixo 'regex:', então seu value é extraído dos dados retornados por esta etapa de acordo com o padrão de expressão regular seguindo o prefixo 'regex:' | |<|nome|x|string|<|Nome da variável.| |<|valor|x|string|<|Valor da variável.| |headers|<|<|<|Elemento raiz para

cabeçalhos HTTP que serão enviados ao realizar uma solicitação. Os cabeçalhos devem ser listados usando a mesma sintaxe que apareceriam no protocolo HTTP. |<|nome|x|string|<|Nome do cabeçalho.| |<|valor|x|string|<|Valor do cabeçalho.| |<|follow_redirects|-|string|0 - NÃO
1 - SIM (padrão)|Siga redirecionamentos HTTP. |<|retrieve_model|-|string|0 - BODY (padrão)
1 - HEADERS
2 - BOTH|Modo de recuperação de resposta HTTP. |<|timeout|-|string|Padrão: 15s|Tempo limite de execução da etapa. Segundos, unidade de tempo com sufixo ou macro de usuário. |<|required|-|string|<|Texto que deve estar presente na resposta. Ignorado se estiver vazio. |<|status_codes|-|string|<|Uma lista delimitada por vírgulas de códigos de status HTTP aceitos. Ignorado se estiver vazio. Por exemplo: 200-201,210-299| |tags|<|-|<|<|Elemento raiz para tags de cenário da web. |<|tag|x|string|<|Nome da tag. |<|valor|-|string|<|Valor da tag. |

Notas de rodapé

¹ Para valores de string, apenas a string será exportada (por exemplo, "ZABBIX_ACTIVE") sem a numeração utilizada nesta tabela. Os números para valores de intervalo (correspondentes aos valores da API) nesta tabela é usado apenas para encomenda.

4 Mapas de rede

Visão geral

O mapa de rede **export** contém:

- todas as imagens relacionadas
- estrutura do mapa - todas as configurações do mapa, todos os elementos contidos com suas configurações, links do mapa e indicadores de status do link do mapa

Quaisquer grupos de hosts, hosts, gatilhos, outros mapas ou outros elementos que possam estar relacionados ao mapa exportado não são exportados. Assim, se faltar pelo menos um dos elementos a que o mapa se refere, importá-lo falhará.

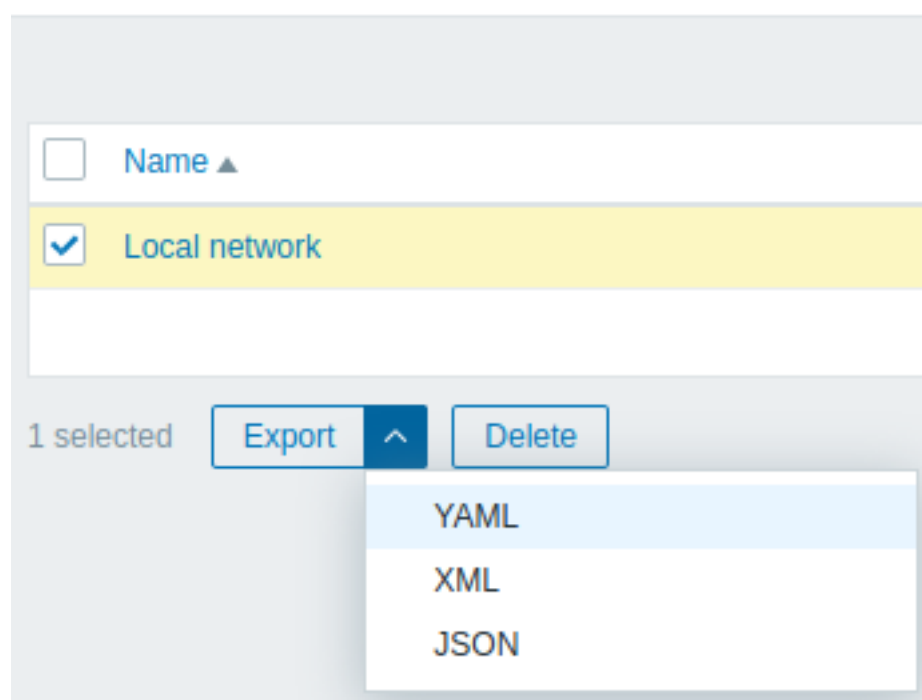
A exportação/importação do mapa de rede é suportada desde o Zabbix 1.8.2.

Exportando

Para exportar mapas de rede, faça o seguinte:

- Vá para: Monitoramento → Mapas
- Marque as caixas de seleção dos mapas de rede para exportar
- Clique em Exportar abaixo da lista

≡ Maps



Dependendo do formato selecionado, os mapas são exportados para um arquivo local com um nome padrão:

- zabbix_export_maps.yaml - na exportação YAML (opção padrão para exportar)
- zabbix_export_maps.xml - na exportação XML

- zabbix_export_maps.json - na exportação JSON

Importando

Para importar mapas de rede, faça o seguinte:

- Vá para: Monitoramento → Mapas
- Clique em Importar à direita
- Selecione o arquivo de importação
- Marque as opções necessárias nas regras de importação
- Clique em Importar

Import

* Import file zbx_export_sysmaps.yaml

Rules	Update existing	Create new	Delete missing
Maps	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Images	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Uma mensagem de sucesso ou falha da importação será exibida na parte dianteira.

Regras de importação:

Regra	Descrição
Atualizar existente	Os mapas existentes serão atualizados com os dados obtidos do arquivo de importação. Caso contrário, eles não serão atualizados.
Criar novo	A importação adicionará novos mapas usando dados do arquivo de importação. Caso contrário, não irá adicioná-los.

Se você desmarcar ambas as opções de mapa e marcar as respectivas opções para imagens, somente as imagens serão importadas. A importação de imagens só está disponível para usuários superadministradores.

Se substituir uma imagem existente, afetará todos os mapas que estão usando esta imagem.

Formato de exportação

Exportar para YAML:

```
zabbix_export:
  versão: '6.0'
  data: '2021-08-31T12:55:10Z'
  imagens:
    -
      nome: Zabbix_server_3D_(128)
      tipo de imagem: '1'
      Imagem codificada: iVBOR...5CYII=
  mapas:
    -
      nome: 'Rede local'
      largura: '680'
      altura: '200'
      label_type: '0'
      label_location: '0'
      destaque: '1'
      expandirproblema: '1'
```

```

elementos de marcação: '1'
show_unack: '0'
gravidade_min: '0'
show_suppressed: '0'
grid_size: '50'
grid_show: '1'
grid_align: '1'
label_format: '0'
label_type_host: '2'
label_type_hostgroup: '2'
label_type_trigger: '2'
label_type_map: '2'
label_type_image: '2'
label_string_host: ''
label_string_hostgroup: ''
label_string_trigger: ''
label_string_map: ''
label_string_image: ''
expand_macros: '1'
fundo: { }
mapa de ícones: { }
URLs: { }
elementos:
-
    tipo de elemento: '0'
    elementos:
    -
        host: 'servidor Zabbix'
        etiqueta: |
            {NOME DE ANFITRIÃO}
            {HOST.CONN}
        label_location: '0'
        x: '111'
        's': '61'
        subtipo de elementos: '0'
        tipo de área: '0'
        largura: '200'
        altura: '200'
        tipo de visualização: '0'
        use_iconmap: '0'
        selementid: '1'
        icon_off:
            nome: Zabbix_server_3D_(128)
        icon_on: { }
        icon_disabled: { }
        icon_maintenance: { }
        URLs: { }
        evaltype: '0'
formas:
-
    tipo: '0'
    x: '0'
    's': '0'
    largura: '680'
    altura: '15'
    texto: '{MAP.NAME}'
    fonte: '9'
    font_size: '11'
    font_color: '000000'
    text_halign: '0'
    text_valign: '0'
    border_type: '0'

```

```

    largura_borda: '0'
    border_color: '000000'
    cor de fundo: ''
    zindex: '0'
  linhas: { }
  ligações: { }

```

Tags de elemento

Os valores de tag de elemento são explicados na tabela abaixo.

Elemento	Propriedade do elemento	Tipo	Intervalo	Descrição
images	nome	string		Elemento raiz para imagens.
	tipo de imagem	inteiro	1 - imagem 2 - plano de fundo	Nome de imagem exclusivo. Tipo de imagem.
	encodedImage			Imagem codificada em Base64.
maps				Elemento raiz para mapas.
	name	string		Nome do mapa exclusivo.
	width	inteiro		Largura do mapa, em pixels.
	altura	inteiro		Altura do mapa, em pixels.
	label_type	integer	0 - label 1 - endereço IP do host 2 - nome do elemento 3 - somente status 4 - nada	Map element label type.
	label_location	integer	0 - inferior 1 - esquerda 2 - direita 3 - superior	Mapeie a localização do rótulo do elemento por padrão.
	destaque	inteiro	0 - não 1 - sim	Ativar destaque de ícone para gatilhos ativos e status de host.
	expandproblem	integer	0 - não 1 - sim	Exibe o gatilho do problema para elementos com um único problema.
	markelements	integer	0 - não 1 - sim	Realçar elementos do mapa que mudaram recentemente de status.
	show_unack	inteiro	0 - contagem de todos os problemas 1 - contagem de problemas não confirmados 2 - contagem de problemas confirmados e não confirmados separadamente	Exibição do problema.

Elemento	Propriedade do elemento	Tipo	Intervalo	Descrição
	severity_min	integer	0 - não classificado 1 - informações 2 - aviso 3 - média 4 - alta 5 - desastre	Severidade mínima do gatilho para mostrar no mapa por padrão.
	show_suppressed	integer	0 - não 1 - yes	Exibe problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.
	grid_size	integer	20, 40, 50, 75 or 100	Tamanho da célula de uma grade de mapa em pixels, if "grid_show=1"
	grid_show	integer	0 - sim 1 - não	Exibe uma grade na configuração do mapa.
	grid_align	integer	0 - sim 1 - não	Alinhar ícones automaticamente na configuração do mapa.
	label_format	integer	0 - não 1 - sim	Usar configuração de etiqueta avançada.
	label_type_host	integer	0 - rótulo 1 - endereço IP do host 2 - nome do elemento 3 - somente status 4 - nada 5 - personalizado label	Exibir como rótulo do host, se "label_format=1"
	label_type_hostgroup	integer	0 - rótulo 2 - nome do elemento 3 - apenas status 4 - nada 5 - rótulo personalizado	Exibir como rótulo do grupo de host, if "label_format=1"
	label_type_trigger	integer	0 - rótulo 2 - nome do elemento 3 - apenas status 4 - nada 5 - rótulo personalizado	Exibir como rótulo de acionador, se "rótulo_formato=1"
	label_type_map	integer	0 - rótulo 2 - nome do elemento 3 - somente status 4 - nada 5 - rótulo personalizado	Exibir como rótulo do mapa, se "rótulo_formato=1"

Elemento	Propriedade do elemento	Tipo	Intervalo	Descrição
urls	label_type_image	integer	0 - rótulo 2 - nome do elemento 4 - nada 5 - rótulo personalizado	Exibir como rótulo de imagem, se "rótulo_format=1"
	label_string_host	string		Rótulo personalizado para elementos host, se "label_type_host=5"
	label_string_hostgroup	string		Rótulo personalizado para elementos do grupo de hosts, se "label_type_hostgroup=5"
	label_string_trigger	string		Rótulo personalizado para elementos acionadores, se "label_type_trigger=5"
	label_string_map	string		Rótulo personalizado para elementos do mapa, se "label_type_map=5"
	label_string_image	string		Rótulo personalizado para elementos de imagem, se "label_type_image=5"
	expandir_macros	inteiro	0 - não 1 - sim	Expandir macros em rótulos na configuração do mapa.
	background	id		ID da imagem de fundo (se houver), if "imagetype=2"
	iconmap	id		ID do mapeamento do ícone (se houver).
	nome	string		Usado por mapas ou cada elemento do mapa. Nome do link.
elementos	url	string		Link URL.
	tipo de elemento	inteiro	0 - host 1 - mapa 2 - gatilho 3 - grupo de hosts 4 - imagem	Tipo de item de mapa ao qual o link pertence.
	tipo de elemento	inteiro	0 - host 1 - mapa 2 - gatilho 3 - grupo de hosts 4 - imagem	tipo de elemento de mapa.
	rótulo	string		Rótulo do ícone.
	label_location	integer	-1 - use map default 0 - bottom 1 - left 2 - right 3 - top	
	x	inteiro		Localização no eixo X.
	y	inteiro		Localização no eixo Y.
	elementsubtype	integer	0 - grupo de host único 1 - todos os grupos de host	Subtipo de elemento, if "elementtype=3"

Elemento	Propriedade do elemento	Tipo	Intervalo	Descrição
tags	areatype	integer	0 - igual ao mapa inteiro 1 - tamanho personalizado	Tamanho da área, if "elementsubtype=1"
	width	inteiro		Largura da área, se "areatype=1"
	altura	inteiro		Altura da área, if "areatype=1"
	viewtype	integer	0 - coloque uniformemente na área	Algoritmo de posicionamento da área, se "elementsubtype=1"
	use_iconmap	integer	0 - não 1 - sim	Use mapeamento de ícones para este elemento. Relevante apenas se o mapeamento de ícones estiver ativado no nível do mapa.
	selementid	id		ID de registro de elemento exclusivo.
elementos	evaltype	integer		Tipo de avaliação para tags. Tags de problema (para elementos de host e grupo de host). Se forem fornecidas tags, apenas problemas com essas tags serão exibidos no mapa.
	tag			Nome da tag.
	valor			Valor da tag.
	operador			Operador.
				Entidades Zabbix que são representadas no mapa (host, host

5 Tipos de mídia

Visão geral

Os tipos de mídia são **exportados** com todos os objetos e relações objetais.

Exportando

Para exportar tipos de mídia, faça o seguinte:

- Vá para: Administração → Tipos de mídia
- Marque as caixas de seleção dos tipos de mídia a serem exportados
- Clique em Exportar abaixo da lista

Media types

<input type="checkbox"/>	Name ▲	Type
<input checked="" type="checkbox"/>	Helpdesk	Webhook

1 selected

Enable **Disable** **Export** **Delete**

- YAML
- XML
- JSON

Dependendo do formato selecionado, os tipos de mídia são exportados para um local arquivo com um nome padrão:

- zabbix_export_mediatypes.yaml - na exportação YAML (opção padrão Para exportar)
- zabbix_export_mediatypes.xml - na exportação XML
- zabbix_export_mediatypes.json - na exportação JSON

Importando

Para importar tipos de mídia, faça o seguinte:

- Vá para: Administração → Tipos de mídia
- Clique em Importar à direita
- Selecione o arquivo de importação
- Marque as opções necessárias nas regras de importação
- Clique em Importar

Import

* Import file **Choose file** zbx_export_mediatypes.yaml

Rules

Update existing Create new Delete missing

Media types ☐ ☒

Import Cancel

Uma mensagem de sucesso ou falha da importação será exibida na parte dianteira.

Regras de importação:

Regra	Descrição
Atualizar existente	Os elementos existentes serão atualizados com os dados retirados do arquivo de importação. Caso contrário, eles não serão atualizados.
Criar novo	A importação adicionará novos elementos usando dados do arquivo de importação. Caso contrário, não irá adicioná-los.

Regra	Descrição
Delete missing	A importação removerá elementos existentes não presentes no arquivo de importação. Caso contrário, não irá removê-los.

Formato de exportação

Exportar para YAML:

```
zabbix_export:
  version: '6.0'
  date: '2021-08-31T12:40:55Z'
  groups:
    - uuid: a571c0d144b14fd4a87a9d9b2aa9fcd6
      name: Templates/Applications
  templates:
    - uuid: 56079badd056419383cc26e6a4fcc7e0
      template: VMware
      name: VMware
      description: |
        You can discuss this template or leave feedback on our forum https://www.zabbix.com/forum/zabbix-s

        Template tooling version used: 0.38
  templates:
    - name: 'VMware macros'
  groups:
    - name: Templates/Applications
  items:
    - uuid: 5ce209f4d94f460488a74a92a52d92b1
      name: 'VMware: Event log'
      type: SIMPLE
      key: 'vmware.eventlog[{$VMWARE.URL},skip]'
      history: 7d
      trends: '0'
      value_type: LOG
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'Collect VMware event log. See also: https://www.zabbix.com/documentation/6.0/manual'
      tags:
        - tag: Application
          value: VMware
    - uuid: ee2edadb8ce943ef81d25dbbba8667a4
      name: 'VMware: Full name'
      type: SIMPLE
      key: 'vmware.fullname[{$VMWARE.URL}]'
      delay: 1h
      history: 7d
      trends: '0'
      value_type: CHAR
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'VMware service full name.'
      preprocessing:
        - type: DISCARD_UNCHANGED_HEARTBEAT
          parameters:
            - 1d
      tags:
        - tag: Application
          value: VMware
    - uuid: a0ec9145f2234fbea79a28c57ebdb44d
      name: 'VMware: Version'
      type: SIMPLE
      key: 'vmware.version[{$VMWARE.URL}]'
      delay: 1h
```

```

history: 7d
trends: '0'
value_type: CHAR
username: '{$VMWARE.USERNAME}'
password: '{$VMWARE.PASSWORD}'
description: 'VMware service version.'
preprocessing:
  - type: DISCARD_UNCHANGED_HEARTBEAT
    parameters:
      - 1d
tags:
  - tag: Application
    value: VMware
discovery_rules:
- uuid: 16ffc933cce74cf28a6edf306aa99782
  name: 'Discover VMware clusters'
  type: SIMPLE
  key: 'vmware.cluster.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Discovery of clusters'
  item_prototypes:
    - uuid: 46111f91dd564a459dbc1d396e2e6c76
      name: 'VMware: Status of "{#CLUSTER.NAME}" cluster'
      type: SIMPLE
      key: 'vmware.cluster.status[{$VMWARE.URL},{#CLUSTER.NAME}]'
      history: 7d
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'VMware cluster status.'
      valuemap:
        name: 'VMware status'
      tags:
        - tag: Application
          value: VMware
- uuid: 8fb6a45cbe074b0cb6df53758e2c6623
  name: 'Discover VMware datastores'
  type: SIMPLE
  key: 'vmware.datastore.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  item_prototypes:
    - uuid: 4b61838ba4c34e709b25081ae5b059b5
      name: 'VMware: Average read latency of the datastore {#DATASTORE}'
      type: SIMPLE
      key: 'vmware.datastore.read[{$VMWARE.URL},{#DATASTORE},latency]'
      history: 7d
      username: '{$VMWARE.USERNAME}'
      password: '{$VMWARE.PASSWORD}'
      description: 'Amount of time for a read operation from the datastore (milliseconds).'
      tags:
        - tag: Application
          value: VMware
    - uuid: 5355c401dc244bc588ccd18767577c93
      name: 'VMware: Free space on datastore {#DATASTORE} (percentage)'
      type: SIMPLE
      key: 'vmware.datastore.size[{$VMWARE.URL},{#DATASTORE},pfree]'
      delay: 5m
      history: 7d
      value_type: FLOAT

```

```

    units: '%'
    username: '{$VMWARE.USERNAME}'
    password: '{$VMWARE.PASSWORD}'
    description: 'VMware datastore space in percentage from total.'
    tags:
      - tag: Application
        value: VMware
- uuid: 84f13c4fde2d4a17baaf0c8c1eb4f2c0
  name: 'VMware: Total size of datastore {#DATASTORE}'
  type: SIMPLE
  key: 'vmware.datastore.size[{$VMWARE.URL},{#DATASTORE}]'
  delay: 5m
  history: 7d
  units: B
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'VMware datastore space in bytes.'
  tags:
    - tag: Application
      value: VMware
- uuid: 540cd0fbc56c4b8ea19f2ff5839ce00d
  name: 'VMware: Average write latency of the datastore {#DATASTORE}'
  type: SIMPLE
  key: 'vmware.datastore.write[{$VMWARE.URL},{#DATASTORE},latency]'
  history: 7d
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Amount of time for a write operation to the datastore (milliseconds).'
  tags:
    - tag: Application
      value: VMware
- uuid: a5bc075e89f248e7b411d8f960897a08
  name: 'Discover VMware hypervisors'
  type: SIMPLE
  key: 'vmware.hv.discovery[{$VMWARE.URL}]'
  delay: 1h
  username: '{$VMWARE.USERNAME}'
  password: '{$VMWARE.PASSWORD}'
  description: 'Discovery of hypervisors.'
  host_prototypes:
    - uuid: 051a1469d4d045cbbf818fcc843a352e
      host: '{#HV.UUID}'
      name: '{#HV.NAME}'
      group_links:
        - group:
            name: Templates/Applications
      group_prototypes:
        - name: '{#CLUSTER.NAME}'
        - name: '{#DATACENTER.NAME}'
      templates:
        - name: 'VMware Hypervisor'
      macros:
        - macro: '{$VMWARE.HV.UUID}'
          value: '{#HV.UUID}'
          description: 'UUID of hypervisor.'
      custom_interfaces: 'YES'
      interfaces:
        - ip: '{#HV.IP}'
- uuid: 9fd559f4e88c4677a1b874634dd686f5
  name: 'Discover VMware VMs'
  type: SIMPLE
  key: 'vmware.vm.discovery[{$VMWARE.URL}]'

```


- NÃO (padrão)
1 - SIM|Se processar as tags retornadas.| |<|show_event_menu|-|string|0 - NO (padrão)
1 - YES|Se {EVENT.TAGS.*} foi resolvido com sucesso em event_menu_url e event_menu_name, este campo indica presença de entrada no menu de eventos.| |<|evento_menu_url|-|string|<|URL da entrada do menu do evento. Suporta macro {EVENT.TAGS.*}.| |<|event_menu_name|-|string|<|Nome da entrada do menu do evento. Suporta macro {EVENT.TAGS.*}.| |parameters|<|<|Elemento raiz para parâmetros de tipo de mídia de webhook.| |<|nome|x|string|<|nome do parâmetro do webhook.| |<|valor|-|string|<|Valor do parâmetro do webhook.|

Notas de rodapé

¹ Para valores de string, apenas a string será exportada (por exemplo, "EMAIL") sem a numeração utilizada nesta tabela. Os números do intervalo valores (correspondentes aos valores da API) nesta tabela é usado para apenas encomendando.

15. Descoberta

Use a barra lateral para acessar o conteúdo da seção Descoberta.

1 descoberta de rede

Visão geral

O Zabbix oferece a funcionalidade de descoberta automática de rede que é eficaz e muito flexível.

Com a descoberta de rede configurada corretamente, você pode:

- acelerar a implantação do Zabbix
- simplificar a administração
- use o Zabbix em ambientes que mudam rapidamente sem excesso de administração

A descoberta de rede Zabbix é baseada nas seguintes informações:

- Faixas de IP
- Disponibilidade de serviços externos (FTP, SSH, WEB, POP3, IMAP, TCP, etc)
- Informações recebidas do agente Zabbix (somente o modo não criptografado é suportado)
- Informações recebidas do agente SNMP

NÃO fornece:

- Descoberta de topologia de rede

A descoberta de rede consiste basicamente em duas fases: descoberta e ações.

Descoberta

O Zabbix verifica periodicamente os intervalos de IP definidos em **descoberta de rede regras**. A frequência do check é configurável para cada regra individualmente.

Observe que uma regra de descoberta sempre será processada por um único processo descobridor. O intervalo de IP não será dividido entre vários processos do descobridor.

Cada regra possui um conjunto de verificações de serviço definidas para serem executadas para o IP variedade.

Note:

As verificações de descoberta são processadas independentemente das outras Verificações. Se alguma verificação não encontrar um serviço (ou falhar), outras verificações serão ainda ser processado.

Toda verificação de um serviço e de um host (IP) realizada pela rede módulo de descoberta gera um evento de descoberta.

Evento	Verificação do resultado do serviço
Serviço Descoberto	O serviço está 'ativo' depois de ter sido 'inativo' ou quando descoberto pela primeira vez.
Service Up	O serviço está 'up', depois que já estava 'up'.
Serviço Perdido	O serviço está 'inativo' depois de estar 'ativo'.
Serviço inativo	O serviço está 'inativo', depois que já estava 'inativo'.
Host Descoberto	Pelo menos um serviço de um host está 'ativo' depois que todos os serviços desse host estavam 'inativos' ou um serviço é descoberto que pertence a um host não registrado.
Host Up	Pelo menos um serviço de um host está 'up', depois que pelo menos um serviço já estava 'up'.

Evento	Verificação do resultado do serviço
Host Lost	Todos os serviços de um host estão 'down' após pelo menos um estar 'up'.
Host Down	Todos os serviços de um host estão 'down', depois de já estarem 'down'.

Ações

Os eventos de descoberta podem ser a base de **ações**, como:

- Envio de notificações
- Adicionando/removendo hosts
- Habilitar/desabilitar hosts
- Adicionando hosts a um grupo
- Removendo hosts de um grupo
- Vinculando hosts a/desvinculando de um modelo
- Execução de scripts remotos

Essas ações podem ser configuradas em relação ao tipo de dispositivo, IP, status, tempo de atividade/tempo de inatividade, etc. Para obter detalhes completos sobre a configuração de ações para eventos baseados em descoberta de rede, consulte [ação operação e condições](#).

Note:

A vinculação de um host descoberto a modelos falhará coletivamente se algum dos modelos vinculáveis tiver uma entidade única (por exemplo, chave de item) que é o mesmo que uma entidade única (por exemplo, chave de item) já existente no host ou em outro dos vinculáveis modelos.

Criação de host

Um host é adicionado se a operação Add host for selecionada. Um anfitrião também adicionado, mesmo se a operação Add host estiver ausente, se você selecionar operações que resultam em ações em um host. Tais operações são:

- habilitar host
- desabilitar host
- adicionar host a um grupo de hosts
- modelo de link para um host

Os hosts criados são adicionados ao grupo Anfitriões descobertos (por padrão, configurável em Administração → Geral → **Outro**). Se você deseja que os hosts sejam adicionados a outro grupo, adicione um Remove from host groups (especificando "Anfitriões descobertos") e também adicionar um Add para hospedar grupos operação (especificando outro grupo de hosts), porque um host deve pertencer a um grupo de hosts.

Nomeação do host

Ao adicionar hosts, um nome de host é o resultado de pesquisa de DNS reverso ou IP endereço se a pesquisa reversa falhar. A pesquisa é realizada a partir do Zabbix servidor ou proxy Zabbix, dependendo de qual está fazendo a descoberta. Se a pesquisa falha no proxy, ela não é repetida no servidor. Se o anfitrião com esse nome já existe, o próximo host obterá **_2** anexado ao nome, depois **_3** e assim por diante.

Também é possível substituir a pesquisa de DNS/IP e usar um item valor para o nome do host, por exemplo:

- Você pode descobrir vários servidores com o agente Zabbix rodando usando um Item do agente Zabbix para descoberta e atribuir nomes próprios a eles automaticamente, com base no valor da string retornado por este item
- Você pode descobrir vários dispositivos de rede SNMP usando um agente SNMP item para descoberta e atribuir nomes próprios a eles automaticamente, com base no valor da string retornado por este item

Se o nome do host tiver sido definido usando um valor de item, ele não será atualizado durante as seguintes verificações de descoberta. Se não for possível definir o host nome usando um valor de item, o valor padrão (nome DNS) é usado.

Se já existir um host com o endereço IP descoberto, um novo host será não criado. No entanto, se a ação de descoberta contiver operações (link modelo, adicionar ao grupo de hosts, etc), eles são executados no hospedeiro.

Remoção de host

Os hosts descobertos por uma regra de descoberta de rede são removidos automaticamente de Monitoramento → Descoberta se uma entidade descoberta não estiver no intervalo de IP da regra. Os hosts são removidos imediatamente.

Criação de interface ao adicionar hosts

Quando hosts são adicionados como resultado da descoberta de rede, eles recebem interfaces criadas de acordo com estas regras:

- os serviços detectados - por exemplo, se uma verificação SNMP for bem-sucedida, um A interface SNMP será criada

- se um host respondeu tanto ao agente Zabbix quanto às solicitações SNMP, ambos tipos de interfaces serão criados
- se os critérios de exclusividade forem o agente Zabbix ou dados retornados por SNMP, o a primeira interface encontrada para um host será criada como padrão. Outros endereços IP serão adicionados como interfaces adicionais.
- se um host respondeu apenas às verificações do agente, ele será criado com um apenas interface do agente. Se ele começar a responder ao SNMP mais tarde, interfaces SNMP adicionais seriam adicionadas.
- se 3 hosts separados foram criados inicialmente, tendo sido descobertos pelos critérios de exclusividade "IP" e, em seguida, a regra de descoberta é modificado para que os hosts A, B e C tenham critérios de exclusividade idênticos resultado, B e C são criados como interfaces adicionais para A, o primeiro hospedeiro. Os hospedeiros individuais B e C permanecem. Em Monitoramento → Discovery as interfaces adicionadas serão exibidas no campo "Discovered dispositivo", em fonte preta e recuada, mas a coluna "Host monitorado" coluna exibirá apenas A, o primeiro host criado. "Uptime/Downtime" não é medido para IPs que são considerados interfaces adicionais.

Alterando a configuração de proxy

Os hosts descobertos por diferentes proxies são sempre tratados como hospedeiros diferentes. Enquanto isso permite realizar a descoberta no IP correspondente intervalos usados por diferentes sub-redes, alterando o proxy para um já sub-rede monitorada é complicado porque as alterações de proxy também devem ser aplicado a todos os hosts descobertos.

Por exemplo, as etapas para substituir o proxy em uma regra de descoberta:

1. desative a regra de descoberta
2. configuração de proxy de sincronização
3. substitua o proxy na regra de descoberta
4. substitua o proxy para todos os hosts descobertos por esta regra
5. habilitar regra de descoberta

1 Configurando uma regra de descoberta de rede

Visão geral

Para configurar uma regra de descoberta de rede usada pelo Zabbix para descobrir hosts e serviços:

- Vá para Configuração → Descoberta
- Clique em Criar regra (ou no nome da regra para editar uma já existente)
- Edite os atributos da regra de descoberta

Atributos da regra

* Name	Local network	
Discovery by proxy	No proxy	
* IP range	192.168.1.1-254	
* Update interval	1h	
* Checks	Type HTTP HTTPS SNMPv2 agent "iso.3 Zabbix agent "system Add	Discovery check Check type SNMPv2 agent * Port range 161 * SNMP community public * SNMP OID iso.3.6.1.2.1.1.1.0
Device uniqueness criteria	<input type="radio"/> IP address <input checked="" type="radio"/> SNMPv2 agent "iso.3.6.1.2.1.1.1.0" <input type="radio"/> Zabbix agent "system.uname"	
Host name	<input type="radio"/> DNS name <input type="radio"/> IP address <input type="radio"/> SNMPv2 agent "iso.3.6.1.2.1.1.1.0" <input checked="" type="radio"/> Zabbix agent "system.uname"	
Visible name	<input checked="" type="radio"/> Host name <input type="radio"/> DNS name <input type="radio"/> IP address <input type="radio"/> SNMPv2 agent "iso.3.6.1.2.1.1.1.0" <input type="radio"/> Zabbix agent "system.uname"	
Enabled	<input checked="" type="checkbox"/>	
	Add	Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	Nome exclusivo da regra. Por exemplo, "Rede local".
Descoberta por proxy	O que realiza a descoberta: sem proxy - o servidor Zabbix está fazendo a descoberta <nome do proxy> - este proxy realiza a descoberta
Intervalo de IP	O intervalo de endereços IP para descoberta. Pode ter os seguintes formatos: IP único: 192.168.1.33 Intervalo de endereços IP: 192.168.1-10.1-255. O intervalo é limitado pelo número total de endereços cobertos (menos de 64 K). Máscara IP: 192.168.4.0/24 máscaras IP compatíveis: /16 - /30 para endereços IPv4 / 112 - /128 para endereços IPv6 Lista: 192.168.1.1-255, 192.168.2.1-100, 192.168.2.200, 192.168.4.0/24 Desde o Zabbix 3.0.0 este campo suporta espaços, tabulação e múltiplas linhas .
Intervalo de atualização	Este parâmetro define com que frequência o Zabbix executará a regra. O intervalo é medido após o término da execução da instância de descoberta anterior, para que não haja sobreposição. Sufixos de tempo são suportados, por exemplo 30s, 1m, 2h, 1d, desde o Zabbix 3.4.0. Macros de usuário são suportados, desde o Zabbix 3.4.0. Observe que se um usuário macro for usada e seu valor for alterado (por exemplo, 1w → 1h), a próxima verificação será executada de acordo com o valor anterior (no futuro com os valores de exemplo).
Verificações	O Zabbix usará esta lista de verificações para descoberta. Clique em Add para configurar uma nova verificação em uma janela pop-up. Verificações compatíveis: SSH, LDAP, SMTP, FTP, HTTP, HTTPS, POP, NNTP, IMAP, TCP, Telnet, agente Zabbix, agente SNMPv1, agente SNMPv2, agente SNMPv3, ping ICMP. Uma descoberta baseada em protocolo usa o net.tcp.service[] funcionalidade para testar cada host, exceto para SNMP que consulta um OID SNMP. O agente Zabbix é testado consultando um item no modo não criptografado. Consulte itens do agente para obter mais detalhes. O parâmetro 'Portas' pode ser um dos seguintes: Porta única: 22 Intervalo de portas: 22-45 Lista: 22-45,55,60-70
Critérios de exclusividade do dispositivo	Os critérios de exclusividade podem ser: endereço IP - sem processamento de vários dispositivos de IP único. Se já existir um dispositivo com o mesmo IP, ele será considerado já descoberto e um novo host não será adicionado. <verificação de descoberta> - verificação do agente Zabbix ou do agente SNMP.
Nome do host	Defina o nome do host técnico de um host criado usando: nome DNS - nome DNS (padrão) endereço IP - endereço IP < verificação de descoberta> - valor de string recebido da verificação de descoberta (por exemplo, agente Zabbix, verificação de agente SNMP) Veja também: Nomeação do host . Esta opção é suportada desde 4.2.0.

Parâmetro	Descrição
Nome visível	<p>Defina o nome do host visível de um host criado usando:</p> <p>Nome do host - nome do host técnico (padrão)</p> <p>nome DNS - nome DNS</p> <p>** Endereço IP** - endereço IP</p> <p><verificação de descoberta> - valor de string recebido da verificação de descoberta (por exemplo, agente Zabbix, verificação de agente SNMP)</p> <p>Veja também: Nomeação do host.</p>
Ativado	<p>Esta opção é suportada desde a versão 4.2.0.</p> <p>Com a caixa marcada a regra está ativa e será executada pelo servidor Zabbix.</p> <p>Se desmarcada, a regra não está ativa. Ele não será executado.</p>

Um cenário da vida real

Neste exemplo, gostaríamos de configurar a descoberta de rede para o local rede com um intervalo de IP de 192.168.1.1-192.168.1.254.

Em nosso cenário queremos:

- descubra os hosts que possuem o agente Zabbix em execução
- execute a descoberta a cada 10 minutos
- adicionar um host ao monitoramento se o tempo de atividade do host for superior a 1 hora
- remover hosts se o tempo de inatividade do host for superior a 24 horas
- adicionar hosts Linux ao grupo "servidores Linux"
- adicionar hosts Windows ao grupo "servidores Windows"
- use o modelo Linux para hosts Linux
- use o modelo Windows para hosts Windows

Passo 1

Definindo uma regra de descoberta de rede para nosso intervalo de IP.

* Name	Local network							
Discovery by proxy	No proxy ▼							
* IP range	192.168.1.1-254							
* Update interval	10m							
* Checks	<table border="1"> <thead> <tr> <th>Type</th> <th>Actions</th> </tr> </thead> <tbody> <tr> <td>Zabbix agent "system.uname"</td> <td>Edit Remove</td> </tr> <tr> <td>Add</td> <td></td> </tr> </tbody> </table>	Type	Actions	Zabbix agent "system.uname"	Edit Remove	Add		
Type	Actions							
Zabbix agent "system.uname"	Edit Remove							
Add								
Device uniqueness criteria	<input checked="" type="radio"/> IP address <input type="radio"/> Zabbix agent "system.uname"							
Host name	<input type="radio"/> DNS name <input type="radio"/> IP address <input checked="" type="radio"/> Zabbix agent "system.uname"							
Visible name	<input checked="" type="radio"/> Host name <input type="radio"/> DNS name <input type="radio"/> IP address <input type="radio"/> Zabbix agent "system.uname"							
Enabled	<input checked="" type="checkbox"/>							

O Zabbix tentará descobrir hosts na faixa de IP de 192.168.1.1-192.168.1.254 conectando-se aos agentes Zabbix e obtendo o valor da chave **system.uname**. O valor recebido do agente pode ser usado para nomear os hosts e também para aplicar diferentes ações para diferentes sistemas operacionais. Por exemplo, vincule servidores Windows ao template Windows, servidores Linux para o template Linux.

A regra será executada a cada 10 minutos.

Quando esta regra for adicionada, o Zabbix iniciará automaticamente a descoberta e geração dos eventos baseados em descoberta para processamento adicional.

Passo 2

Definindo uma **ação** de descoberta para adicionando os servidores Linux descobertos ao respectivo grupo/modelo.

Action	Operations										
* Name	Add discovered Linux servers										
Type of calculation	And ▼ A and B and C and D										
Conditions	<table border="1"> <thead> <tr> <th>Label</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Received value contains <i>Linux</i></td> </tr> <tr> <td>B</td> <td>Discovery status equals <i>Up</i></td> </tr> <tr> <td>C</td> <td>Service type equals <i>Zabbix agent</i></td> </tr> <tr> <td>D</td> <td>Uptime/Downtime is greater than or equals 3600</td> </tr> </tbody> </table> Add	Label	Name	A	Received value contains <i>Linux</i>	B	Discovery status equals <i>Up</i>	C	Service type equals <i>Zabbix agent</i>	D	Uptime/Downtime is greater than or equals 3600
Label	Name										
A	Received value contains <i>Linux</i>										
B	Discovery status equals <i>Up</i>										
C	Service type equals <i>Zabbix agent</i>										
D	Uptime/Downtime is greater than or equals 3600										

A ação será ativada se:

- o serviço "agente Zabbix" está "ativo"
- o valor de system.uptime (a chave do agente Zabbix que usamos na regra definição) contém "Linux"
- O tempo de atividade é de 1 hora (3600 segundos) ou mais

Action	Operations
Default subject	Discovery: {DISCOVERY.DEVICE.STATUS}, {DISCOVERY.DEVICE.IPADDRESS}
Default message	Discovery rule: {DISCOVERY.RULE.NAME} Device IP: {DISCOVERY.DEVICE.IPADDRESS} Device DNS: {DISCOVERY.DEVICE.DNS} Device status: {DISCOVERY.DEVICE.STATUS} Device uptime: {DISCOVERY.DEVICE.uptime} Device service name: {DISCOVERY.SERVICE.NAME}
Operations	Details Add to host groups: Linux servers Link to templates: Linux Add

A ação executará as seguintes operações:

- adicione o host descoberto ao grupo "servidores Linux" (e também adicione host se não foi adicionado anteriormente)
- vincule o host ao modelo Linux. Zabbix irá iniciar automaticamente monitorando o host usando itens e gatilhos do "Linux" modelo.

Etapas 3

Definir uma ação de descoberta para adicionar os servidores Windows descobertos a o respectivo grupo/modelo.

Action	Operations										
* Name	Add discovered Windows servers										
Type of calculation	<div>And</div> <div>A and B and C and D</div>										
Conditions	<table border="1"> <thead> <tr> <th>Label</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Received value contains <i>Windows</i></td> </tr> <tr> <td>B</td> <td>Discovery status equals <i>Up</i></td> </tr> <tr> <td>C</td> <td>Service type equals <i>Zabbix agent</i></td> </tr> <tr> <td>D</td> <td>Uptime/Downtime is greater than or equals 3600</td> </tr> </tbody> </table> Add	Label	Name	A	Received value contains <i>Windows</i>	B	Discovery status equals <i>Up</i>	C	Service type equals <i>Zabbix agent</i>	D	Uptime/Downtime is greater than or equals 3600
Label	Name										
A	Received value contains <i>Windows</i>										
B	Discovery status equals <i>Up</i>										
C	Service type equals <i>Zabbix agent</i>										
D	Uptime/Downtime is greater than or equals 3600										

Action	Operations
Default subject	Discovery: {DISCOVERY.DEVICE.STATUS}, {DISCOVERY.DEVICE.IPADDRESS}
Default message	Discovery rule: {DISCOVERY.RULE.NAME} Device IP: {DISCOVERY.DEVICE.IPADDRESS} Device DNS: {DISCOVERY.DEVICE.DNS} Device status: {DISCOVERY.DEVICE.STATUS} Device uptime: {DISCOVERY.DEVICE.UPTIME} Device service name: {DISCOVERY.SERVICE.NAME}
Operations	Details Add to host groups: Windows servers Link to templates: Windows Add

Passo 4

Definindo uma ação de descoberta para remover servidores perdidos.

Action	Operations								
* Name	Remove lost servers								
Type of calculation	And ▼ A and B and C								
Conditions	<table border="1"> <thead> <tr> <th>Label</th> <th>Name</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>Uptime/Downtime is greater than or equals 86400</td> </tr> <tr> <td>B</td> <td>Discovery status equals Down</td> </tr> <tr> <td>C</td> <td>Service type equals Zabbix agent</td> </tr> </tbody> </table> Add	Label	Name	A	Uptime/Downtime is greater than or equals 86400	B	Discovery status equals Down	C	Service type equals Zabbix agent
Label	Name								
A	Uptime/Downtime is greater than or equals 86400								
B	Discovery status equals Down								
C	Service type equals Zabbix agent								

Action	Operations				
Default subject	Discovery: {DISCOVERY.DEVICE.STATUS}, {DISCOVERY.DEVICE.IPADDRESS}				
Default message	Discovery rule: {DISCOVERY.RULE.NAME} Device IP: {DISCOVERY.DEVICE.IPADDRESS} Device DNS: {DISCOVERY.DEVICE.DNS} Device status: {DISCOVERY.DEVICE.STATUS} Device uptime: {DISCOVERY.DEVICE.UPTIME} Device service name: {DISCOVERY.SERVICE.NAME}				
Operations	<table border="1"> <thead> <tr> <th>Details</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Remove host</td> <td> Edit Remove </td> </tr> </tbody> </table> Add	Details	Action	Remove host	Edit Remove
Details	Action				
Remove host	Edit Remove				

Um servidor será removido se o serviço "agente Zabbix" estiver 'inativo' por mais mais de 24 horas (86400 segundos).

2 Registro automático de agente ativo

Visão geral

É possível permitir o registro automático do agente Zabbix ativo, após qual o servidor pode começar a monitorá-los. Desta forma, novos hospedeiros podem ser adicionados para monitoramento sem configurá-los manualmente no servidor.

O registro automático pode acontecer quando um agente ativo anteriormente desconhecido solicita para cheques.

O recurso pode ser muito útil para monitoramento automático de novos Cloud nós. Assim que você tiver um novo node no Cloud Zabbix irá iniciar automaticamente a coleta de dados de desempenho e disponibilidade do hospedeiro.

O registro automático de agente ativo também suporta o monitoramento de hosts com verificações passivas. Quando o agente ativo pede cheques, desde que tenha os parâmetros de configuração 'ListenIP' ou 'ListenPort' definidos no arquivo de configuração, estes são enviados ao servidor. (Se vários endereços IP forem especificados, o primeiro será enviado para o servidor.)

Servidor, ao adicionar o novo host registrado automaticamente, usa o IP recebido endereço e porta para configurar o agente. Se nenhum valor de endereço IP for recebido, aquele usado para a conexão de entrada é usado. Se nenhuma porta valor é recebido, 10050 é usado.

É possível especificar que o host deve ser registrado automaticamente com um **Nome DNS** como o agente padrão interface.

O registro automático é executado novamente:

- se as informações do host **metadados** forem alteradas:
 - devido a HostMetadata alterado e agente reiniciado
 - devido ao valor retornado pelo HostMetadataItem alterado
- para hosts criados manualmente com metadados ausentes
- se um host for alterado manualmente para ser monitorado por outro Zabbix procurador
- se o autoregistro para o mesmo host vier de um novo proxy Zabbix

Configuração

Especificar servidor

Certifique-se de ter o servidor Zabbix identificado no agente **arquivo de configuração** - zabbix_agentd.conf

ServidorAtivo=10.0.0.1

A menos que você defina especificamente um Hostname em zabbix_agentd.conf, o nome do host do sistema do local do agente será usado pelo servidor para nomear o hospedeiro. O nome do host do sistema no Linux pode ser obtido executando o comando 'nome do host'.

Se Hostname estiver definido na configuração do agente Zabbix como um lista de hosts delimitada por vírgulas, serão criados hosts para todos os nomes de host.

Reinicie o agente depois de fazer qualquer alteração no arquivo de configuração.

Ação para registro automático de agente ativo

Quando o servidor recebe uma solicitação de registro automático de um agente, ele chama uma **ação**. Uma ação de evento source "Autoregistration" deve ser configurado para autorregistro do agente.

Note:

A configuração de **network discovery** não é necessário ter o auto-registro de agentes ativos.

No frontend do Zabbix, vá para Configuration → Actions, selecione Autoregistration como origem do evento e clique em Create action:

- Na guia Ação, dê um nome à sua ação
- Opcionalmente especificar **condições**. Você pode fazer uma correspondência de substring ou uma correspondência de expressão regular no condições para o nome do host/metadados do host. Se você for usar o Condição "Metadados do host", consulte a próxima seção.
- Na guia Operações, adicione operações relevantes, como - 'Adicionar host', 'Adicionar ao grupo de hosts' (por exemplo, Anfitriões descobertos), 'Link para modelos', etc.

Se os hosts que serão registrados automaticamente ser suportado apenas para monitoramento ativo (como hosts que firewall do seu servidor Zabbix), então você pode querer criar um modelo específico como Template_Linux-active para vincular.

Os hosts criados são adicionados ao grupo Anfitriões descobertos (por padrão, configurável em Administração → Geral → **Outro**). Se você deseja que os hosts sejam adicionados a outro grupo, adicione um Remove from host group (especificando "hosts descobertos") e também adicione um Add to operação host group (especificando outro grupo de hosts), porque um host deve pertencer a um grupo de hosts.

Registro automático seguro

Uma maneira segura de registro automático é possível configurando autenticação com conexões criptografadas.

O nível de criptografia é configurado globalmente em Administration → **Geral**, na seção Registro automático acessível através do menu suspenso para o certo. É possível selecionar sem criptografia, criptografia TLS com PSK autenticação ou ambos (para que alguns hosts possam se registrar sem criptografia enquanto outros por criptografia).

A autenticação por PSK é verificada pelo servidor Zabbix antes de adicionar um host. Se for bem-sucedido, o host será adicionado e **Conexões de/para host** são definidos como 'PSK' somente com identidade/chave pré-compartilhada igual ao registro automático global contexto.

::: não importante Para garantir a segurança do registro automático em instalações usando proxies, criptografia entre servidor Zabbix e proxy deve ser habilitado. :::

Usando DNS como interface padrão

HostInterface e HostInterfaceItem **configuração parâmetros** permitem especificar um valor personalizado para a interface do host durante o registro automático.

Mais especificamente, eles são úteis se o host deve ser registrado automaticamente com um nome DNS como interface de agente padrão em vez de seu IP Morada. Nesse caso, o nome DNS deve ser especificado ou retornado como o valor dos parâmetros `HostInterface` ou `HostInterfaceItem`. Observação que se o valor de um dos dois parâmetros mudar, o a interface de host registrada automaticamente é atualizada. Então é possível atualizar a interface padrão para outro nome DNS ou atualize-a para um endereço IP. No entanto, para que as alterações entrem em vigor, o agente deve ser reiniciado.

Note:

Se os parâmetros `HostInterface` ou `HostInterfaceItem` não forem configurado, o parâmetro `listen_dns` é resolvido a partir do endereço IP. Se tal resolução estiver configurada incorretamente, ela pode quebrar registro automático devido a nome de host inválido.

Usando metadados do host

Quando o agente está enviando uma solicitação de registro automático para o servidor, ele envia seu nome de host. Em alguns casos (por exemplo, nós de nuvem da Amazon) um nome de host não é suficiente para o servidor Zabbix diferenciar os hosts descobertos. Hospedeiro metadados podem ser usados opcionalmente para enviar outras informações de um agente ao servidor.

Os metadados do host são configurados no agente **configuration arquivo** - `zabbix_agentd.conf`. Existem 2 maneiras de especificar metadados de host no arquivo de configuração:

`HostMetadados`

`HostMetadataItem`

Veja a descrição das opções no link acima.

::: não importante Uma tentativa de registro automático acontece toda vez que um agente ativo envia uma solicitação para atualizar as verificações ativas para o servidor. O o atraso entre as solicitações é especificado no parâmetro **RefreshActiveChecks** do agente. A primeira solicitação é enviada imediatamente após o agente ser reiniciado. :::

Exemplo 1

Usando metadados de host para distinguir entre hosts Linux e Windows.

Digamos que você gostaria que os hosts fossem registrados automaticamente pelo servidor Zabbix. Você tem agentes Zabbix ativos (veja a seção "Configuração" acima) em sua rede. Existem hosts Windows e hosts Linux em sua rede e você tem "Linux by Zabbix agent" e "Windows by Zabbix agent" templates disponíveis em seu frontend Zabbix. Assim, no registro do host, você gostaria que o modelo Linux/Windows apropriado fosse aplicado a o host que está sendo registrado. Por padrão, apenas o nome do host é enviado para o servidor no registro automático, o que pode não ser suficiente. Para fazer certifique-se de que o modelo adequado seja aplicado ao host, você deve usar o host metadados.

Configuração de front-end

A primeira coisa a fazer é configurar o frontend. Crie 2 ações. A primeira ação:

- Nome: registro automático do host Linux
- Condições: Os metadados do host contêm Linux
- Operações: Link para templates: Linux

Note:

Você pode pular uma operação "Adicionar host" neste caso. Vinculação para um modelo requer a adição de um host primeiro para que o servidor faça isso automaticamente.

A segunda ação:

- Nome: registro automático do host do Windows
- Condições: os metadados do host contêm Windows
- Operações: Link para templates: Windows

Configuração do agente

Agora você precisa configurar os agentes. Adicione a próxima linha ao agente arquivos de configuração:

`HostMetadataItem=system.uname`

Dessa forma, você garante que os metadados do host contenham "Linux" ou "Windows" dependendo do host em que um agente está sendo executado. Um exemplo de anfitrião metadados neste caso:

Linux: Linux server3 3.2.0-4-686-pae #1 SMP Debian 3.2.41-2 i686 GNU/Linux

Windows: Windows WIN-OPXGGSTYNH0 6.0.6001 Windows Server 2008 Service Pack 1 Intel IA-32

Não se esqueça de reiniciar o agente depois de fazer qualquer alteração no arquivo de configuração.

Exemplo 2

Passo 1

Usando metadados de host para permitir alguma proteção básica contra anfitriões registrando.

Configuração de front-end

Crie uma ação no frontend, usando algum código secreto difícil de adivinhar para não permitir hosts indesejados:

- Nome: Ação de registro automático Linux
- Condições:
 - * Tipo de cálculo: E
 - * Condição (A): Os metadados do host contêm //Linux//
 - * Condição (B): os metadados do host contêm //21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
- * Operações:
 - * Enviar mensagem aos usuários: Admin por todas as mídias
 - * Adicionar a grupos de hosts: servidores Linux
 - * Link para modelos: Linux

Observe que este método sozinho não fornece proteção forte porque os dados são transmitidos em texto simples. A recarga do cache de configuração é necessário para que as alterações tenham efeito imediato.

Configuração do agente

Adicione a próxima linha ao arquivo de configuração do agente:

```
HostMetadata=Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

onde "Linux" é uma plataforma, e o resto da string é o texto secreto difícil de adivinhar.

Não se esqueça de reiniciar o agente depois de fazer qualquer alteração no arquivo de configuração.

Passo 2

É possível adicionar monitoramento adicional para um já cadastrado hospedeiro.

Configuração de front-end

Atualize a ação no frontend:

- Nome: Ação de registro automático Linux
- Condições:
 - * Tipo de cálculo: E
 - * Condição (A): Os metadados do host contêm Linux
 - * Condição (B): Os metadados do host contêm 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
- * Operações:
 - * Enviar mensagem aos usuários: Admin por todas as mídias
 - * Adicionar a grupos de hosts: servidores Linux
 - * Link para modelos: Linux
 - * Link para templates: MySQL por Zabbix Agent

Configuração do agente

Atualize a próxima linha no arquivo de configuração do agente:

```
HostMetadata=MySQL no Linux 21df83bf21bf0be663090bb8d4128558ab9b95fba66a6dbf834f8b91ae5e08ae
```

Não se esqueça de reiniciar o agente depois de fazer qualquer alteração no arquivo de configuração.

3 Descoberta de baixo nível

Visão geral A descoberta de baixo nível fornece uma maneira de criar itens automaticamente, gatilhos e gráficos para diferentes entidades em um computador. Por exemplo, O Zabbix pode iniciar automaticamente o monitoramento de sistemas de arquivos ou rede interfaces em sua máquina, sem a necessidade de criar itens para cada sistema de arquivos ou interface de rede manualmente. Além disso, é possível configurar o Zabbix para remover entidades desnecessárias automaticamente com base em resultados reais da descoberta realizada periodicamente.

Um usuário pode definir seus próprios tipos de descoberta, desde que siga um determinado protocolo JSON.

A arquitetura geral do processo de descoberta é a seguinte.

Primeiro, um usuário cria uma regra de descoberta em "Configuração" → "Modelos" → Coluna "Descoberta". Uma regra de descoberta consiste em (1) um item que descobre as entidades necessárias (por exemplo, sistemas de arquivos ou rede interfaces) e (2) protótipos de itens, gatilhos e gráficos que deve ser criado com base no valor desse item.

Um item que descobre as entidades necessárias é como um item normal visto em outro lugar: o servidor pede a um agente Zabbix (ou qualquer que seja o tipo de o item está definido como) para um valor desse item, o agente responde com um valor textual. A diferença é que o valor com o qual o agente responde deve conter uma lista de entidades descobertas em um formato JSON. Enquanto o detalhes deste formato são importantes apenas para implementadores de verificações de descoberta, é necessário saber que o valor retornado contém uma lista de macro → pares de valores. Por exemplo, artigo "net.if.discovery" pode retornar dois pares: "{#IFNAME}" → "lo" e "{#IFNAME}" → "eth0".

Essas macros são usadas em nomes, chaves e outros campos de protótipo onde eles são então substituídos pelos valores recebidos para criar itens, gatilhos, gráficos ou mesmo hosts para cada entidade descoberta. Ver a lista completa de [opções](#) para usar macros LLD.

Quando o servidor recebe um valor para um item de descoberta, ele verifica o macro → pares de valor e para cada par gera itens reais, gatilhos, e gráficos, com base em seus protótipos. No exemplo com "net.if.discovery" acima, o servidor geraria um conjunto de itens, gatilhos e gráficos para a interface de loopback "lo", e outro conjunto para a interface "eth0".

Observe que desde o **Zabbix 4.2**, o formato do JSON retornado por as regras de descoberta de baixo nível foram alteradas. Não é mais esperado que o JSON conterá o objeto "data". A descoberta de baixo nível agora aceita um JSON normal contendo um array, para dar suporte a novos recursos como o pré-processamento do valor do item e caminhos personalizados para valores de macro de descoberta de baixo nível em um documento JSON.

As chaves de descoberta integradas foram atualizadas para retornar uma matriz de linhas LLD na raiz do documento JSON. Zabbix irá extrair automaticamente uma macro e valor se um campo de matriz usar a sintaxe {#MACRO} como chave. Qualquer novo as verificações de descoberta nativa usarão a nova sintaxe sem os "dados" elementos. Ao processar primeiro um valor de descoberta de baixo nível, a raiz é localizado (array em \$. ou \$.data).

Embora o elemento "data" tenha sido removido de todos os itens nativos relacionados para descoberta, para compatibilidade com versões anteriores, o Zabbix ainda aceitará o Notação JSON com um elemento "data", embora seu uso seja desencorajado. Se o JSON contém um objeto com apenas um elemento de matriz "data", então ele irá extrair automaticamente o conteúdo do elemento usando JSONPath \$.dados. A descoberta de baixo nível agora aceita LLD opcional definido pelo usuário macros com um caminho personalizado especificado na sintaxe JSONPath.

Como resultado das mudanças acima, os agentes mais novos não mais será capaz de trabalhar com um servidor Zabbix mais antigo.

Veja também: [Entidades descobertas](#)

Configurando a descoberta de baixo nível Ilustraremos a descoberta de baixo nível com base em um exemplo de arquivo descoberta do sistema.

Para configurar a descoberta, faça o seguinte:

- Vá para: Configuração → Modelos ou Hosts
- Clique em Discovery na linha de um modelo/host apropriado

≡ Templates

<input type="checkbox"/>	Name ▲	Hosts	Applications	Items	Triggers	Graphs	Dashboards	Discovery
<input type="checkbox"/>	Linux OS agent	Hosts 1	Applications 11	Items 42	Triggers 14	Graphs 8	Dashboards 1	Discovery 3

- Clique em Criar regra de descoberta no canto superior direito da tela
- Preencha o formulário de regra de descoberta com os detalhes necessários

Regra de descoberta

O formulário de regra de descoberta contém cinco guias, representando, da esquerda para certo, o fluxo de dados durante a descoberta:

- Regra de descoberta - especifica, mais importante, o item interno ou script personalizado para recuperar dados de descoberta
- Pré-processamento - aplica algum pré-processamento aos dados descobertos
- Macros LLD - permite extrair alguns valores de macro para usar em itens descobertos, gatilhos, etc.
- Filters - permite filtrar os valores descobertos
- Substitui - permite modificar itens, gatilhos, gráficos ou host protótipos ao aplicar a objetos descobertos específicos

A guia **Regra de descoberta** contém a chave do item a ser usada para descoberta (assim como alguns atributos gerais da regra de descoberta):

Discovery rule
Preprocessing
LLD macros
Filters 4
Overrides

* Name
Mounted filesystem discovery

Type
Zabbix agent

* Key
vfs.fs.discovery

* Update interval
1h

Custom intervals

Type
Interval
Period

Flexible
Scheduling
50s
1-7,00:00-24:00

Add

* Keep lost resources period
30d

Description
Discovery of file systems of different types.

Enabled
☒

Add
Test
Cancel

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Parâmetro	Descrição
Nome	Nome da regra de descoberta.
Type	O tipo de verificação para realizar a descoberta. Neste exemplo, estamos usando uma chave de item do agente Zabbix. A regra de descoberta também pode ser um item dependente , dependendo de um item normal. Não pode depender de outra regra de descoberta. Para um item dependente, selecione o respectivo tipo (Item dependente) e especifique o item mestre no campo 'Item mestre'. O item mestre deve existir.
Key	Insira a chave do item de descoberta (até 2.048 caracteres). Por exemplo, você pode usar a chave de item "vfs.fs.discovery" integrada para retornar um JSON com a lista de sistemas de arquivos presentes no computador e seus tipos. Observe que outra opção para descoberta do sistema de arquivos é usar os resultados da descoberta pela chave de agente "vfs.fs.get", suportada desde o Zabbix 4.4.5 (veja exemplo).

Parâmetro	Descrição
Intervalo de atualização	<p>Este campo especifica com que frequência o Zabbix realiza a descoberta. No início, quando você está apenas configurando a descoberta do sistema de arquivos, você pode querer configurá-lo para um pequeno intervalo, mas quando souber que funciona, você pode configurá-lo para 30 minutos ou mais, porque os sistemas de arquivos geralmente não mudam com muita frequência .</p> <p>Sufixos de tempo são suportados, por exemplo 30s, 1m, 2h, 1d, desde o Zabbix 3.4.0.</p> <p>Macros de usuário são suportados, desde o Zabbix 3.4.0.</p> <p>Observação: O intervalo de atualização só pode ser definido como '0' se existirem intervalos personalizados com um valor diferente de zero. Se definido como '0' e existir um intervalo personalizado (flexível ou programado) com um valor diferente de zero, o item será pesquisado durante a duração do intervalo personalizado.</p> <p>Observe que, para uma regra de descoberta existente, a descoberta pode ser executado imediatamente pressionando o botão Check now (#form_buttons).</p>
Intervalos personalizados	<p>Você pode criar regras personalizadas para verificar o item:</p> <p>Flexível - cria uma exceção para o Intervalo de atualização (intervalo com frequência diferente)</p> <p>Agendamento - cria um agendamento de pesquisa personalizado.</p> <p>Para obter informações detalhadas, consulte Intervalos personalizados. O agendamento é suportado desde o Zabbix 3.0.0.</p>
Período de manutenção de recursos perdidos	<p>Este campo permite especificar a duração por quanto tempo a entidade descoberta será retida (não será excluída) quando seu status de descoberta se tornar "Não mais descoberta" (entre 1 hora a 25 anos; ou "0").</p> <p>Sufixos de tempo são suportados, por exemplo 2h, 1d, desde o Zabbix 3.4.0.</p> <p>Macros de usuário são suportados, desde o Zabbix 3.4.0.</p> <p>Nota: Se definido como "0", entidades serão excluídas imediatamente. O uso de "0" não é recomendado, pois apenas editar erroneamente o filtro pode acabar na entidade que está sendo excluída com todos os dados históricos.</p>
Descrição	Digite uma descrição.
Ativado	Se marcado, a regra será processada.

Note:

O histórico da regra de descoberta não é preservado.

Pré-processando

A guia **Pré-processamento** permite definir as regras de transformação a serem aplicadas ao resultado da descoberta. Uma ou várias transformações são possíveis nesta etapa. As transformações são executadas na ordem em que são definidos. Todo o pré-processamento é feito pelo servidor Zabbix.

Veja também:

- [Pré-processando detalhes](#)
- [Teste de pré-processamento](#)

Preprocessing steps

Name

Parameters



1:

Regular expression



pattern



2:

JSONPath



\$.pool

Add

Tipo	Transformação	Descrição
Texto	Expressão regular	<p>Faça corresponder o valor recebido à expressão regular <pattern> e substitua o valor pela <saída> extraída. A expressão regular oferece suporte à extração de no máximo 10 grupos capturados com a sequência \N.</p> <p>Parâmetros:</p> <p>padrão - expressão regular</p> <p>saída - modelo de formatação de saída.</p> <p>Uma sequência de escape \N (onde N=1...9) é substituída pelo enésimo grupo correspondente. Uma sequência de escape \0 é substituída pelo texto correspondente.</p> <p>Se você marcar a caixa de seleção Personalizado em caso de falha, é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir um mensagem de erro especificada.</p>
	Substituir	<p>Encontre a string de pesquisa e substitua-a por outra (ou nada). Todas as ocorrências da string de pesquisa serão substituídas.</p> <p>Parâmetros:</p> <p>search string - a string para localizar e substituir, diferencia maiúsculas de minúsculas (obrigatório)</p> <p>replacement - a string para substitua a string de pesquisa por. A string de substituição também pode estar vazia, permitindo excluir a string de pesquisa quando encontrada.</p> <p>É possível usar sequências de escape para procurar ou substituir quebras de linha, retorno de carro, tabulações e espaços "\n \r \t\s"; a barra invertida pode ser escapada como "\\" e as sequências de escape podem ser escapadas como "\\n". O escape de quebras de linha, retorno de carro, guias é feito automaticamente durante a descoberta de baixo nível.</p> <p>Suportado desde 5.0.0.</p>

Dados
estruturados

Tipo	Transformação	Descrição
	JSONPath	<p>Extraia valor ou fragmento de dados JSON usando funcionalidade JSONPath.</p> <p>Se você marcar a caixa de seleção Personalizado em caso de falha, o item não será tornam-se incompatíveis em caso de falha na etapa de pré-processamento e é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir uma mensagem de erro especificada.</p>
	XML XPath	<p>Extraia valor ou fragmento de dados XML usando a funcionalidade XPath.</p> <p>Para que esta opção funcione, o servidor Zabbix deve ser compilado com suporte a libxml.</p> <p>Exemplos:</p> <p><code>number(/document /item/value)</code> extrairá 10 de <code><document><item><value>10</value></item></do</code> <code>number(/document/item/@attribute)</code> extrairá 10 de <code><document><item</code> <code>attribute="10"></item></document></code> <code>/document/item</code> extrairá <code><item><value>10</value ></item></code> de <code><document><item><value>10</value></item></do</code></p> <p>Observe que namespaces não são suportados.</p> <p>Suportado desde 4.4.0.</p> <p>Se você marcar a caixa de seleção Personalizado em caso de falha, é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir uma mensagem de erro especificada.</p>
	CSV para JSON	<p>Converter dados do arquivo CSV para o formato JSON.</p> <p>Para obter mais informações, consulte: CSV para pré-processamento JSON.</p> <p>Suportado desde 4.4.0.</p>
	XML para JSON	<p>Converter dados em formato XML para JSON.</p> <p>Para obter mais informações, consulte: Regras de serialização.</p> <p>Se você marcar a caixa de seleção Personalizado em caso de falha, é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir uma mensagem de erro especificada.</p>
Scripts personalizados	JavaScript	<p>Insira o código JavaScript no bloco que aparece ao clicar no campo de parâmetro ou em Abrir.</p> <p>Observe que o comprimento do JavaScript disponível depende do banco de dados usado.</p> <p>Para obter mais informações, consulte: Pré-processamento de Javascript</p>
Validação		

Tipo	Transformação	Descrição
	Não corresponde a expressão regular	<p>Especifique uma expressão regular que um valor não deve corresponder.</p> <p>Ex. <code>Error: (.*)\.</code></p> <p>Se você marcar a caixa de seleção Custom on fail, é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir um erro especificado mensagem.</p>
	Verifique se há erro no JSON	<p>Verifique se há uma mensagem de erro no nível do aplicativo localizada em JSONpath. Interrompa o processamento se for bem-sucedido e a mensagem não estiver vazia; caso contrário, continue o processamento com o valor que estava antes dessa etapa de pré-processamento. Observe que esses erros de serviço externo são relatados ao usuário como estão, sem adicionar informações da etapa de pré-processamento.</p> <p>Por exemplo, <code>\$.errors</code>. Se um JSON como <code>{"errors": "e1"}</code> for recebido, a próxima etapa de pré-processamento não será executada.</p> <p>Se você marcar a caixa de seleção Personalizado em caso de falha, é possível especificar opções personalizadas de tratamento de erros: para descartar o valor, definir um valor especificado ou definir uma mensagem de erro especificada.</p>
	Verifique se há erro no XML	<p>Verifique se há uma mensagem de erro no nível do aplicativo localizada em xpath. Interrompa o processamento se for bem-sucedido e a mensagem não estiver vazia; caso contrário, continue o processamento com o valor que estava antes dessa etapa de pré-processamento. Observe que esses erros de serviço externo são relatados ao usuário como estão, sem adicionar informações da etapa de pré-processamento.</p> <p>Nenhum erro será relatado em caso de falha ao analisar XML inválido.</p> <p>Suportado desde a versão 4.4.0.</p> <p>Se você marque a caixa de seleção Personalizado em caso de falha, é possível especificar opções personalizadas de tratamento de erros: descartar o valor, definir um valor especificado ou definir uma mensagem de erro especificada.</p>

Estrangulamento

Tipo	Transformação	Descrição
	Descartar inalterado com pulsação	<p>Descartar um valor se ele não foi alterado dentro do período de tempo definido (em segundos).</p> <p>Valores inteiros positivos são suportados para especificar os segundos (mínimo - 1 segundo). Sufixos de tempo podem ser usados neste campo (por exemplo, 30s, 1m, 2h, 1d). Macros de usuário e macros de descoberta de baixo nível podem ser usadas neste campo.</p> <p>Apenas uma opção de limitação pode ser especificada para um item de descoberta. Por exemplo, 1m. Se um texto idêntico for passado para esta regra duas vezes em 60 segundos, ele será descartado.</p> <p>Observação: A alteração de protótipos de itens não redefine a limitação. A limitação é redefinida somente quando as etapas de pré-processamento são alteradas.</p>
Prometeu	Prometheus para JSON	<p>Converter as métricas necessárias do Prometheus para JSON.</p> <p>Consulte verificações do Prometheus para obter mais detalhes.</p>

Observe que se a regra de descoberta foi aplicada ao host via template então o conteúdo desta guia é somente leitura.

Macros personalizadas

A guia **macros LLD** permite especificar a descoberta personalizada de baixo nível macros.

As macros personalizadas são úteis nos casos em que o JSON retornado não tem as macros necessárias já definidas. Assim, por exemplo:

- A chave nativa `vfs.fs.discovery` para descoberta do sistema de arquivos retorna um JSON com algumas macros LLD predefinidas, como `{#FSNAME}`, `{#FSTYPE}`. Essas macros podem ser usadas no item, acionar protótipos (veja as seções subsequentes da página) diretamente; definindo costume macros não são necessárias;
- O item do agente `vfs.fs.get` também retorna um JSON com `filesystem dados`, mas sem nenhuma macro LLD predefinida. Neste caso você pode definir as macros você mesmo e mapeie-as para os valores no JSON usando Caminho JSON:

LLD macro	JSONPath
{#FSNAME}	\$.fsname
{#FSTYPE}	\$.fstype

Add

Os valores extraídos podem ser usados em itens descobertos, gatilhos, etc. Observe que os valores serão extraídos do resultado da descoberta e qualquer etapas de pré-processamento até agora.

Parâmetro	Descrição
Macro LLD	Nome da macro de descoberta de baixo nível, usando a seguinte sintaxe: <code>{#MACRO}</code> .

Parâmetro	Descrição
JSONPath	<p>Caminho usado para extrair o valor da macro LLD de uma linha LLD, usando a sintaxe JSONPath.</p> <p>Por exemplo, \$.foo extrairá "bar" e "baz" deste JSON: [{"foo": "bar"}, {"foo": "baz"}]</p> <p>Os valores extraídos do JSON retornado são usados para substituir as macros LLD nos campos de item, gatilho, etc. de protótipo.</p> <p>JSONPath pode ser especificado usando a notação de ponto ou a notação de colchetes. A notação de colchetes deve ser usada no caso de quaisquer caracteres especiais e Unicode, como \$['unicode + caracteres especiais #1']['unicode + caracteres especiais #2'].</p>

Filtro

Um filtro pode ser usado apenas para gerar itens reais, gatilhos e gráficos para entidades que atendem aos critérios. A guia **Filtros** contém Definições de filtro de regra de descoberta que permitem filtrar valores de descoberta:

Discovery rule Preprocessing LLD macros **Filtros 4** Overrides

Type of calculation And (A and B) and (C and D)

Filters	Label	Macro		Regular expression
A		{#FSNAME}	matches	{\$VFS.FS.FSNAME.MATCH
B		{#FSNAME}	does not match	{\$VFS.FS.FSNAME.NOT_M
C		{#FSTYPE}	matches	{\$VFS.FS.FSTYPE.MATCH
D		{#FSTYPE}	does not match	{\$VFS.FS.FSTYPE.NOT_M

Add

Parâmetro	Descrição
Tipo de cálculo	<p>As seguintes opções para calcular filtros estão disponíveis:</p> <p>E - todos os filtros devem ser passados;</p> <p>Ou - suficiente se um filtro for passado;</p> <p>And/Or - usa And com nomes de macro diferentes e Or com o mesmo nome de macro;</p> <p>Expressão personalizada - oferece a possibilidade de definir um cálculo personalizado de filtros. A fórmula deve incluir todos os filtros da lista. Limitado a 255 símbolos.</p>

Parâmetro	Descrição
Filtros	<p>Os seguintes operadores de condição de filtro estão disponíveis: corresponde, não corresponde, existe, não existe.</p> <p>Os operadores Corresponde e não corresponde esperam um Perl Expressão regular compatível (PCRE). Por exemplo, se você estiver interessado apenas em sistemas de arquivos C:, D: e E:, você pode colocar {#FSNAME} em "Macro" e "^C ^D ^E" regular expressão em campos de texto "Expressão regular". A filtragem também é possível por tipos de sistema de arquivos usando a macro {#FSTYPE} (por exemplo, "^ext ^reiserfs") e por tipos de unidade (suportado apenas pelo agente do Windows) usando a macro {#FSDRIVETYPE} (por exemplo, "fixed").</p> <p>Você pode inserir uma expressão regular ou fazer referência a uma expressão regular global no campo "Expressão regular".</p> <p>Para testar uma expressão regular, você pode usar "grep -E", por exemplo:</p> <pre>for f in ext2 nfs reiserfs smbfs; faça eco \$f \ grep -E '^ex'</pre> <p>macro no Windows é suportada desde o Zabbix 3.0.0.</p> <p>Os operadores Exists e not exist permitem filtrar entidades com base na presença ou ausência da macro LLD especificada na resposta (com suporte desde a versão 5.4.0).</p> <p>A definição de vários filtros é compatível desde o Zabbix 2.4.0. Observe que, se uma macro do filtro estiver ausente na resposta, a entidade encontrada será ignorada, a menos que uma condição "não existe" seja especificada para essa macro.</p>

Um erro ou um erro de digitação na expressão regular usada na regra LLD (por exemplo, um "sistemas de arquivos para descoberta" incorreto expressão regular) pode causar a exclusão de milhares de configurações elementos, valores históricos e eventos para muitos anfitriões.

::: não importante O banco de dados Zabbix no MySQL deve ser criado como diferencia maiúsculas de minúsculas se nomes de sistema de arquivos que diferem apenas por maiúsculas e minúsculas devem ser descoberto corretamente. :::

Sobrepôr

A guia **Substituir** permite definir regras para modificar a lista de itens, protótipos de gatilho, gráfico e host ou seus atributos para descobertas objetos que atendem a determinados critérios.

Discovery rule
Preprocessing
LLD macros
Filters
Overrides 1

Overrides

Name	Stop processing	Action
1: Set trigger with threshold of 50%	Yes	Remove

Add

As substituições (se houver) são exibidas em uma lista reordenável de arrastar e soltar e executados na ordem em que são definidos. Para configurar detalhes de uma nova substituição, clique em [Add](#) nas Substituições bloquear. Para editar uma substituição existente, clique no nome da substituição. Um pop-up será aberta a janela permitindo editar os detalhes da regra de substituição.

Override

* Name

Set trigger with threshold of 50%

If filter matches

Continue overrides

Stop processing

Filters

	Label Macro		Regular expression
A	{#FSNAME}	matches	^Vtmp\$
Add			

Operations

Condition

Trigger prototype does not equal *Disk space is low (used > 50%)*

[Add](#)

Todos os parâmetros obrigatórios estão marcados com asteriscos vermelhos.

Parâmetro	Descrição
Nome	Um nome de substituição exclusivo (por regra LLD).
Se o filtro corresponder	Define se as próximas substituições devem ser processadas quando as condições do filtro forem atendidas: Continuar substituições - as substituições subsequentes serão processadas. Parar processamento - operações de (se houver) e essa substituição será executada, as substituições subsequentes serão ignoradas para linhas LLD correspondentes.
Filtros	Determina a quais entidades descobertas a substituição deve ser aplicada. Os filtros de substituição são processados após a regra de descoberta filtros e têm a mesma funcionalidade.
Operações	As operações de substituição são exibidas com estes detalhes: Condição - um tipo de objeto (protótipo de item/protótipo de gatilho/protótipo de gráfico/protótipo de host) e uma condição a ser atendida (igual/diferente /contém/não contém/corresponde/não corresponde) Ação - são exibidos links para editar e remover uma operação.

Configurando uma operação

Para configurar detalhes de uma nova operação, clique em [Add](#) em Operações bloquear. Para editar uma operação existente, clique em [Edit](#) ao lado do Operação. Uma janela pop-up onde você pode editar os detalhes da operação será abrir.

New operation

Object Trigger prototype ▼

Condition does not equal ▼ Disk space is low (used > 50%)

Create enabled ☐ Original

Discover ☒ Yes No

Severity ☐ Original

Tags ☐ Original

Add

Parâmetro	Descrição
Objeto	Quatro tipos de objetos estão disponíveis: Protótipo de item Protótipo de gatilho Protótipo de gráfico Protótipo de host
Condição	Permite filtrar entidades às quais a operação deve ser aplicada.
Operador	Operadores suportados: igual - aplica-se a este protótipo não é igual - aplica-se a todos os protótipos, exceto este contém* * - aplicar, se o nome do protótipo contiver esta string não contém* - aplicar, se o nome do protótipo não contiver esta string corresponde - aplicar, se o nome do protótipo corresponder à expressão regular >não corresponde - aplica-se, se o nome do protótipo não corresponder à expressão regular
Padrão	Uma expressão regular ou uma string para pesquisar.
Objeto: Pro- tótipo de item Criar ha- bili- tado	Quando a caixa de seleção estiver marcada, os botões aparecerão, permitindo substituir as configurações originais do protótipo do item: Sim - o item será adicionado em um estado habilitado. >Não - o item será adicionado a uma entidade descoberta, mas em um estado desabilitado.
Descobrir	Quando a caixa de seleção estiver marcada, os botões aparecerão, permitindo substituir as configurações originais do protótipo do item: Sim - o item será adicionado. Não - o item não será adicionado.

Parâmetro	Descrição
Intervalo de atualização	<p>Quando a caixa de seleção estiver marcada, duas opções aparecerão, permitindo definir um intervalo diferente para o item:</p> <p>Atraso - Intervalo de atualização do item. Macros do usuário e sufixos de tempo (por exemplo, 30s, 1m, 2h, 1d) são suportados. Deve ser definido como 0 se Intervalo personalizado for usado.</p> <p>Intervalo personalizado - clique em Add para especificar flexível /programação de intervalos. Para obter informações detalhadas, consulte Intervalos personalizados.</p>
Período de armazenamento do histórico	<p>Quando a caixa de seleção estiver marcada, os botões aparecerão, permitindo definir diferentes períodos de armazenamento do histórico para o item:</p> <p>Não manter histórico - se selecionado, o histórico não será armazenado.</p> <p>Período de armazenamento - se selecionado, um campo de entrada para especificar o período de armazenamento aparecerá à direita. Macros de usuário e macros LLD são suportados.</p>
Período de armazenamento de tendências	<p>Quando a caixa de seleção estiver marcada, os botões aparecerão, permitindo definir diferentes períodos de armazenamento de tendências para o item:</p> <p>Não manter tendências - se selecionado, as tendências não serão armazenadas.</p> <p>Período de armazenamento - se selecionado, um campo de entrada para especificar o período de armazenamento aparecerá à direita. Macros de usuário e macros LLD são suportados.</p>
Tags	Quando a caixa de seleção estiver marcada, um novo bloco aparecerá

Botões de formulário

Os botões na parte inferior do formulário permitem realizar várias operações.

Add	Adicione uma regra de descoberta. Este botão está disponível apenas para novas regras de descoberta.
Update	Atualize as propriedades de uma regra de descoberta. Este botão está disponível apenas para regras de descoberta existentes.
Clone	Crie outra regra de descoberta com base nas propriedades da regra de descoberta atual.
Check now	<p>Execute a descoberta com base na regra de descoberta imediatamente. A regra de descoberta já deve existir. Consulte mais detalhes.</p> <p>Observe que ao realizar a descoberta imediatamente, o cache de configuração não é atualizado, portanto, o resultado não refletirá alterações muito recentes na configuração da regra de descoberta.</p>
Delete	Excluir a regra de descoberta.

Cancel

Cancelar a edição das propriedades da regra de descoberta.


Entidades descobertas As capturas de tela abaixo ilustram como os itens descobertos, acionadores e gráficos se parecem com a configuração do host. As entidades descobertas são prefixado com um link laranja para uma regra de descoberta de onde eles vêm.

Items				
All hosts / Remote proxy: New host Enabled ZBX SNMP JMX IPMI Applications 11 Items 41 T				
<input type="checkbox"/>	Wizard	Name	Triggers	Key
<input type="checkbox"/>	...	Mounted filesystem discovery: Free disk space on / (percentage)	Triggers 1	vfs.fs.size[/,pfr
<input type="checkbox"/>	...	Mounted filesystem discovery: Used disk space on /		vfs.fs.size[/,use
<input type="checkbox"/>	...	Mounted filesystem discovery: Free disk space on /		vfs.fs.size[/,free
<input type="checkbox"/>	...	Mounted filesystem discovery: Free inodes on / (percentage)	Triggers 1	vfs.fs.inode[/,p

Observe que as entidades descobertas não serão criadas caso haja entidades já existentes com os mesmos critérios de exclusividade, por exemplo, um item com a mesma chave ou gráfico com o mesmo nome. Um erro mensagem é exibida neste caso no frontend que o nível baixo regra de descoberta não pôde criar determinadas entidades. A regra de descoberta em si, no entanto, não ficará sem suporte porque alguma entidade não pôde ser criado e teve que ser ignorado. A regra de descoberta continuará criar/atualizar outras entidades.

Itens (da mesma forma, gatilhos e gráficos) criados por uma descoberta de baixo nível regra será excluída automaticamente se uma entidade descoberta (sistema de arquivos, interface, etc) deixa de ser descoberto (ou não passa no filtro mais). Neste caso, os itens, gatilhos e gráficos serão excluídos após os dias definidos no campo Manter o período de recursos perdidos.

Quando as entidades descobertas se tornam 'Não mais descobertas', uma vida inteira indicador é exibido na lista de itens. Mova o ponteiro do mouse sobre ele e uma mensagem será exibida indicando quantos dias faltam para o item é excluído.

1m	7d	1y	Zabbix agent	Enabled	
The item is not discovered anymore and will be deleted in 29d 23h 44m (on 2015-08-31 at 23:27).					

Se as entidades foram marcadas para exclusão, mas não foram excluídas no tempo esperado (regra de descoberta desabilitada ou host de item), eles serão excluído na próxima vez que a regra de descoberta for processada.

Entidades contendo outras entidades, marcadas para exclusão, serão não atualizar se alterado no nível da regra de descoberta. Por exemplo, Os acionadores baseados em LLD não serão atualizados se contiverem itens marcados para exclusão.

Triggers			Group	all
All hosts / Remote proxy: New host Enabled ZBX SNMP JMX IPMI Applications 11 Items 41 T				
<input type="checkbox"/>	Severity	Name ▲		
<input type="checkbox"/>	Warning	Mounted filesystem discovery: Free disk space is less than 20% on volume /		
<input type="checkbox"/>	Warning	Mounted filesystem discovery: Free inodes is less than 20% on volume /		

Graphs

Group

[All hosts](#) / [Remote proxy: New host](#)
Enabled
ZBX
SNMP
JMX
IPMI
Applications 11
Items 41
T

<input type="checkbox"/>	Name ▲
<input type="checkbox"/>	Template OS Linux: CPU jumps
<input type="checkbox"/>	Template OS Linux: CPU load
<input type="checkbox"/>	Template OS Linux: CPU utilization
<input type="checkbox"/>	Mounted filesystem discovery: Disk space usage /

Outros tipos de descoberta Mais detalhes e instruções sobre outros tipos de descoberta pronta para uso são disponível nas seguintes seções:

- descoberta de [rede interfaces](#);
- descoberta de [CPUs e CPU núcleos](#);
- descoberta de [SNMP OIDs](#);
- descoberta de [JMX objetos](#);
- descoberta usando [ODBC SQL consultas](#);
- descoberta de [Windows serviços](#);
- descoberta de [host interfaces](#) em Zabbix.

Para obter mais detalhes sobre o formato JSON para itens de descoberta e um exemplo de como implementar seu próprio descobridor de sistema de arquivos como um script Perl, consulte [criando regras de LLD personalizadas](#).

1 Protótipos de itens

Assim que uma regra for criada, vá para os itens dessa regra e pressione "Criar protótipo de item" para criar um protótipo de item. Observe como a macro `{#FSNAME}` é usado onde um nome de sistema de arquivos é necessário. Quando a regra de descoberta é processado, esta macro será substituída pelo arquivo descoberto sistema.

Item prototype
Tags
Preprocessing

* Name

{#FSNAME}: Used space

Type

Zabbix agent

* Key

vfs.fs.size[{#FSNAME},used]

Sel

Type of information

Numeric (unsigned)

Units

B

* Update interval

1m

Custom intervals

Type	Interval	Period
Flexible	Scheduling	50s
		1-7,00:00-24:00

Add

* History storage period

Do not keep history

Storage period

7d

* Trend storage period

Do not keep trends

Storage period

365d

Value mapping

type here to search

Sel

Description

Used storage in Bytes

Create enabled

☒

Discover

☒

Add

Test

Cancel

Descoberta de baixo nível **macros** e usuário **macros** podem ser usados na configuração do protótipo do item e no pré-processamento do valor do item **parâmetros**. Observação que quando usado em intervalos de atualização, uma única macro tem que preencher todo o campo. Várias macros em um campo ou macros misturadas com texto não são suportado.

Note:

O escape específico do contexto de macros de descoberta de baixo nível é realizado para uso seguro em expressão regular e pré-processamento XPath parâmetros.

Atributos específicos para protótipos de itens:

Parâmetro	Descrição
Criar habilitado	Se marcado, o item será adicionado em um estado habilitado. Se desmarcado, o item será adicionado a uma entidade descoberta, mas em um estado desabilitado.

Parâmetro	Descrição
Discover	Se marcado (padrão), o item será adicionado a uma entidade descoberta. Se desmarcado, o item não será adicionado a uma entidade descoberta, a menos que esta configuração seja substituída na regra de descoberta.

Podemos criar vários protótipos de itens para cada métrica de sistema de arquivos que estamos interessado em:

≡ Item prototypes

All templates / Template Module Windows filesystem...
Discovery list / Mounted filesystem discovery

Item prototypes 3
Trigger prototypes 2
Graph prototypes 1
Host prototypes

<input type="checkbox"/>	Name ▲	Key	Interval
<input type="checkbox"/> ...	{#FSNAME}: Space utilization	vfs.fs.size[{#FSNAME},pused]	1m
<input type="checkbox"/> ...	{#FSNAME}: Total space	vfs.fs.size[{#FSNAME},total]	1m
<input type="checkbox"/> ...	{#FSNAME}: Used space	vfs.fs.size[{#FSNAME},used]	1m

0 selected
Create enabled
Create disabled
Mass update
Delete

Clique no ícone de três pontos para abrir o menu do protótipo de item específico com estas opções:
 - Criar protótipo de gatilho - crie um protótipo de gatilho com base neste protótipo de item - Protótipos de gatilho - clique para ver uma lista com links para protótipos de gatilho já configurados deste protótipo de item - Criar item dependente - criar um item dependente para este protótipo de item

Atualização em massa a opção está disponível se você quiser atualizar as propriedades de vários itens protótipos de uma só vez.

2 Protótipos de gatilho

Criamos protótipos de gatilhos de maneira semelhante aos protótipos de itens:

Trigger prototype Dependencies

* Name	Free disk space is less than 20% on volume {#FSNAME}
--------	--

Severity	Not classified	Information	Warning	Average	High	Critical
----------	----------------	-------------	---------	---------	------	----------

* Expression	{Template OS Linux:vfs.fs.size[{#FSNAME},pfree].last(0)}<20
--------------	---

Expression constructor

[illegible]

PROBLEM event generation mode	Single	Multiple
-------------------------------	--------	----------

OK event closes	All problems	All problems if tag values match
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Tags

tag

value


Add

Allow manual close ☐

URL

Description

Create enabled ☒

Discover 

Add

Cancel

Parâmetro

Descrição

Se marcado, o gatilho será adicionado em um estado ativado.
Se desmarcado, o gatilho será adicionado a uma entidade descoberta, mas em um estado desativado.

Parâmetro	Descrição
Discover	<p>Se marcado (padrão), o gatilho será adicionado a uma entidade descoberta.</p> <p>Se desmarcado, o gatilho não será adicionado a uma entidade descoberta, a menos que essa configuração seja substituída na regra de descoberta.</p>

Quando gatilhos reais são criados a partir dos protótipos, pode ser necessário ser flexível quanto a qual constante ('20' em nosso exemplo) é usada para comparação na expressão. Veja como **macros de usuário com context** pode ser útil para alcançar tal flexibilidade.

Você pode definir **dependências** entre protótipos de trigger também (suportado desde o Zabbix 3.0). Façam isso, vá para a aba Dependências. Um protótipo de gatilho pode depender outro protótipo de gatilho da mesma regra de descoberta de baixo nível (LLD) ou em um gatilho regular. Um protótipo de gatilho pode não depender de um gatilho protótipo de uma regra LLD diferente ou em um gatilho criado a partir do gatilho protótipo. O protótipo de gatilho de host não pode depender de um gatilho de um modelo.

Trigger prototypes

[All templates](#) / [Template OS Linux](#)
[Discovery list](#) / [Mounted filesystem discovery](#)
[Item prototypes](#) 5

<input type="checkbox"/>	SEVERITY	NAME ▲	EXPRESSION
<input type="checkbox"/>	Warning	Free disk space is less than 20% on volume {#FSNAME}	{Template OS
<input type="checkbox"/>	Warning	Free inodes is less than 20% on volume {#FSNAME}	{Template OS

3 Protótipos de gráficos

Também podemos criar protótipos de gráficos:

Graph prototype
Preview

* Name

{#FSNAME}: Disk space usage

* Width

600

* Height

340

Graph type

Pie

Show legend

☒

3D view

☒

* Items

	Name	Type
1:	Template Module Linux filesystems by Zabbix agent: {#FSNAME}: Total space	Graph
2:	Template Module Linux filesystems by Zabbix agent: {#FSNAME}: Used space	Simple

Add
Add prototype

Discover

☒

Add

Cancel

Atributos específicos para protótipos de gráfico:

Parâmetro	Descrição
Discover	<p>Se marcado (padrão), o gráfico será adicionado a uma entidade descoberta.</p> <p>Se desmarcado, o gráfico não será adicionado a uma entidade descoberta, a menos que essa configuração seja substituída na regra de descoberta.</p>

Graph prototypes		
All templates / Template OS Linux Discovery list / Mounted filesystem discovery Item prototypes 5		
<input type="checkbox"/>	NAME ▲	WIDTH
<input type="checkbox"/>	Disk space usage {#FSNAME}	600

Por fim, criamos uma regra de descoberta que se parece com a mostrada abaixo. Isto tem cinco protótipos de itens, dois protótipos de gatilho e um gráfico protótipo.

Discovery rules

All templates / Template Module Linux filesystems...			Items	Triggers	Graphs	Dashboards	Discovery rules
<input type="checkbox"/>	Template	Name ▲	Items				
<input type="checkbox"/>	Template Module Linux filesystems by Zabbix agent	Mounted filesystem discovery	Item prototypes 4				

Observação: Para configurar protótipos de host, consulte a seção sobre [host prototype](#) configuração em monitoramento de máquina virtual.

4 Notas sobre descoberta de baixo nível

Usando macros LLD em contextos de macro de usuário

As macros LLD podem ser usadas dentro do contexto da macro do usuário, por exemplo, [em acionar protótipos](#).

Várias regras LLD para o mesmo item

Desde a versão 3.2 do agente Zabbix é possível definir várias regras de descoberta de baixo nível com o mesmo item de descoberta.

Para fazer isso você precisa definir o agente Alias [parâmetro](#), permitindo usar chaves de item de descoberta alteradas em diferentes regras de descoberta, por exemplo `vfs.fs.discovery[foo]`, `vfs.fs.discovery[bar]`, etc.

Limites de dados para valores de retorno

Não há limite para dados JSON de regra de descoberta de baixo nível se for recebido diretamente pelo servidor Zabbix, pois os valores de retorno são processados sem ser armazenado em um banco de dados. Também não há limite para personalização regras de descoberta de baixo nível, no entanto, se pretende adquirir Dados LLD usando um parâmetro do usuário e, em seguida, o valor de retorno do parâmetro do usuário limite se aplica (512 KB).

Se os dados tiverem que passar pelo proxy Zabbix, eles devem armazenar esses dados em banco de dados assim [banco de dados limites](#) se aplicam.

5 Regras de descoberta

Use a barra lateral para ver exemplos de configuração de regras de descoberta para vários casos.

1 Descoberta de sistemas de arquivos montados

Visão geral

É possível descobrir sistemas de arquivos montados e suas propriedades (nome do ponto de montagem, tipo de ponto de montagem, tamanho do sistema de arquivos e inode Estatísticas).

Para fazer isso, você pode usar uma combinação de:

- o item do agente `vfs.fs.get` como o item mestre
- regra de descoberta de baixo nível dependente e protótipos de itens

Configuração

Item mestre

Crie um item de agente Zabbix usando a seguinte chave:

`vfs.fs.get`

Item	Tags	Preprocessing
<div>* Name <input type="text" value="vfs.fs.get item"/></div>		
<div>Type <input type="text" value="Zabbix agent"/></div>		
<div>* Key <input type="text" value="vfs.fs.get"/></div>		
<div>* Host interface <input type="text" value="127.0.0.1 : 10050"/></div>		
<div>Type of information <input type="text" value="Text"/></div>		

Defina o tipo de informação como "Texto" para dados JSON possivelmente grandes.

Os dados retornados por este item conterão algo como o seguinte para um sistema de arquivos montado:

```
{
  "fsname": "/",
  "fstype": "rootfs",
  "bytes": {
    "total": 1000,
    "grátis": 500,
    "usado": 500,
    "pgrátis": 50,00,
    "empurrado": 50,00
  },
  "inodos": {
    "total": 1000,
    "grátis": 500,
    "usado": 500,
    "pgrátis": 50,00,
    "empurrado": 50,00
  }
}
```

Regra LLD dependente

Crie uma regra de descoberta de baixo nível como tipo "Item dependente":

Discovery rule	Preprocessing	LLD macros	Filters	Overrides
<div>* Name <input type="text" value="Discovery rule for vfs.fs.get"/></div>				
<div>Type <input type="text" value="Dependent item"/></div>				
<div>* Key <input type="text" value="fs.mountpoint.discovery"/></div>				
<div>* Master item <input type="text" value="Zabbix server: vfs.fs.get item"/></div>				
<div>* Keep lost resources period <input type="text" value="30d"/></div>				

Como item mestre selecione o item `vfs.fs.get` que criamos.

Na aba "Macros LLD" defina macros personalizadas com o correspondente Caminho JSON:

Discovery rule	Preprocessing	LLD macros 2	Filters	Overrides
LLD macros				
LLD macro		JSONPath		
<input data-bbox="560 423 1227 486" type="text" value="{#FSNAME}"/>		<input data-bbox="1244 423 1477 486" type="text" value="\$.fsname"/>		
<input data-bbox="560 512 1227 575" type="text" value="{#FSTYPE}"/>		<input data-bbox="1244 512 1477 575" type="text" value="\$.fstype"/>		
Add				

Protótipo de item dependente

Crie um protótipo de item com o tipo "Item dependente" nesta regra LLD. Como item mestre para este protótipo selecione o item `vfs.fs.get` que criamos.

Item prototype	Tags	Preprocessing
* Name <input data-bbox="654 969 1453 1028" type="text" value="Free disk space on {#FSNAME}, type: {#FSTYPE}"/>		
Type <input data-bbox="654 1059 1034 1120" type="text" value="Dependent item"/>		
* Key <input data-bbox="654 1151 1453 1209" type="text" value="free[{#FSNAME}]"/>		
* Master item <input data-bbox="654 1240 1453 1299" type="text" value="Zabbix server: vfs.fs.get item"/>		
Type of information <input data-bbox="654 1330 1048 1388" type="text" value="Numeric (unsigned)"/>		

Observe o uso de macros personalizadas no nome e na chave do protótipo do item:

- Nome: Espaço livre em disco em `{#FSNAME}`, digite: `{#FSTYPE}`
- Chave: `free[{#FSNAME}]`

Como tipo de informação, use:

- Numérico (sem sinal) para métricas como 'gratuito', 'total', 'usado'
- Numeric (float) para métricas como 'pfree', 'pused' (porcentagem)

Na guia "Pré-processamento" do protótipo do item, selecione JSONPath e use o seguinte expressão JSONPath como parâmetro:

`$. [?(@.fsname=='{#FSNAME}')].bytes.free.first()`

Item prototype	Tags	Preprocessing 1
Preprocessing steps		
Name		
Parameters		
1: <input data-bbox="534 1962 863 2016" type="text" value="JSONPath"/>		
<input data-bbox="879 1962 1477 2016" type="text" value="\$. [?(@.fsname=='{#FSNAME}')].bytes.free.first()"/>		
Add		

Quando a descoberta for iniciada, será criado um item por cada ponto de montagem. Este item retornará o número de bytes livres para o ponto de montagem fornecido.

2 Descoberta de interfaces de rede

De maneira semelhante ao [arquivo sistemas](#) são descobertos, é possível também descobrir interfaces de rede.

Chave do item

A chave do item a ser usada na [descoberta regra](#) é

`net.if.discovery`

Este item é suportado desde o agente Zabbix 2.0.

Macros compatíveis

Você pode usar a macro `{#IFNAME}` na regra de descoberta [filtro](#) e protótipos de itens, gatilhos e gráficos.

Exemplos de protótipos de itens que você pode querer criar com base em "net.if.discovery":

- "net.if.in[{#IFNAME},bytes]",
- "net.if.out[{#IFNAME},bytes]".

Observe que no Windows `{#IFGUID}` também é retornado.

3 Descoberta de CPUs e núcleos de CPU

De maneira semelhante ao [arquivo sistemas](#) são descobertos, é possível descobrir também CPUs e núcleos de CPU.

Chave do item

A chave do item a ser usada na [descoberta regra](#) é

`system.cpu.discovery`

Este item é suportado desde o agente Zabbix 2.4.

Macros compatíveis

Esta chave de descoberta retorna duas macros - `{#CPU.NUMBER}` e `{#CPU.STATUS}` identificando o número e o status do pedido da CPU, respectivamente. Observe que uma distinção clara não pode ser feita entre reais, físicos e processadores, núcleos e hyperthreads. `{#CPU.STATUS}` no Linux, UNIX e Os sistemas BSD retornam o status do processador, que pode ser "online" ou "offline". Em sistemas Windows, esta mesma macro pode representar um terceiro valor - "desconhecido" - que indica que um processador foi detectado, mas nenhuma informação foi coletada para ele ainda.

A descoberta de CPU depende do processo coletor do agente para permanecer consistente com os dados fornecidos pelo coletor e economizar recursos em obtenção dos dados. Isso faz com que esta chave de item não funcione com o sinalizador de linha de comando `test (-t)` do binário do agente, que retornar um status `NOT_SUPPORTED` e uma mensagem que o acompanha indicando que o processo de coleta não foi iniciado.

Os protótipos de itens que podem ser criados com base na descoberta da CPU incluem, por exemplo, exemplo:

- `system.cpu.util[{#CPU.NUMBER},<tipo>,<modo>]`
- `system.hw.cpu[{#CPU.NUMBER},<info>]`

Para obter uma descrição detalhada da chave do item, consulte [item do agente Zabbix keys](#).

4 Descoberta de OIDs SNMP

Visão geral

Nesta seção vamos realizar um SNMP [discovery](#) em um switch.

Chave do item

Ao contrário da descoberta de sistema de arquivos e interface de rede, o item não necessariamente tem que ter uma chave "snmp.discovery" - tipo de item de SNMP agente é suficiente.

A descoberta de OIDs SNMP é suportada desde o servidor/proxy Zabbix 2.0.

Para configurar a regra de descoberta, faça o seguinte:

- Vá para: Configuração → Modelos

- Clique em Discovery na linha de um modelo apropriado

Templates

☐ Name ▲

☐ Interfaces SNMP

☐ Hosts

☐ Items

☐ Triggers

☐ Graphs

☐ Dashboards

☐ Discovery

- Clique em Criar regra de descoberta no canto superior direito da tela
- Preencha o formulário de regra de descoberta com os detalhes necessários como no captura de tela abaixo

Discovery rule

Preprocessing

LLD macros

Filters 12

Overrides

* Name

Network interfaces discovery

Type

SNMP agent

* Key

net.if.discovery

* SNMP OID

discovery[#{#IFALIAS},1.3.6.1.2.1.31.1.1.1.18,#{#IFNAME},1.3.6.1.2.1.31.1.1.1.1,#{#IF

* Update interval

1h

Custom intervals

Type

Interval

Period

Flexible

Scheduling

50s

1-7,00:00-24:00

Add

* Keep lost resources period

30d

Description

Discovering interfaces from IF-MIB.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Os OIDs a serem descobertos são definidos no campo SNMP OID nas seguintes formato: `discovery[#{#MACRO1}, oid1, #{#MACRO2}, oid2, ...]`

onde `{#MACRO1}`, `{#MACRO2}` ... são nomes válidos de macro lld e `oid1`, `oid2`... são OIDs capazes de gerar valores significativos para esses macros. Uma macro interna `{#SNMPINDEX}` contendo o índice do OID descoberto é aplicado a entidades descobertas. O descoberto as entidades são agrupadas pelo valor da macro `{#SNMPINDEX}`.

Para entender o que queremos dizer, vamos realizar alguns snmpwalks em nosso switch:

```
$ snmpwalk -v 2c -c public 192.168.1.1 IF-MIB::ifDescr
IF-MIB::ifDescr.1 = STRING: WAN
IF-MIB::ifDescr.2 = STRING: LAN1
IF-MIB::ifDescr.3 = STRING: LAN2
```

```
$ snmpwalk -v 2c -c public 192.168.1.1 IF-MIB::ifPhysAddress
IF-MIB::ifPhysAddress.1 = STRING: 8:0:27:90:7a:75
IF-MIB::ifPhysAddress.2 = STRING: 8:0:27:90:7a:76
IF-MIB::ifPhysAddress.3 = STRING: 8:0:27:2b:af:9e
```

E defina SNMP OID para: `discovery[#{#IFDESCR}, ifDescr, #{#IFPHYSADDRESS}, ifPhysAddress]`

Agora esta regra descobrirá entidades com macros `{#IFDESCR}` definidas como **WAN**, **LAN1** e **LAN2**, macros `{#IFPHYSADDRESS}` definidas como **8:0:27:90:7a:75**, **8:0:27:90:7a:76** e **8:0:27:2b:af:9e**, macros `{#SNMPINDEX}` definidas para os índices de OIDs descobertos **1**, **2** e **3**:

```
[
  {
    "#{#SNMPINDEX}": "1",
```

```

    "{#IFDESCR}": "WAN",
    "{#IFPHYSADDRESS}": "8:0:27:90:7a:75"
  },
  {
    "{#SNMPINDEX}": "2",
    "{#IFDESCR}": "LAN1",
    "{#IFPHYSADDRESS}": "8:0:27:90:7a:76"
  },
  {
    "{#SNMPINDEX}": "3",
    "{#IFDESCR}": "LAN2",
    "{#IFPHYSADDRESS}": "8:0:27:2b:af:9e"
  }
]

```

Se uma entidade não tiver o OID especificado, então o macro será omitida para esta entidade. Por exemplo, se tivermos o seguintes dados:

```

ifDescr.1 "Interface #1"
ifDescr.2 "Interface #2"
ifDescr.4 "Interface #4"

ifAlias.1 "eth0"
ifAlias.2 "eth1"
ifAlias.3 "eth2"
ifAlias.5 "eth4"

```

Então, neste caso, a descoberta de SNMP discovery[{#IFDESCR}, ifDescr, {#IFALIAS}, ifAlias] retornará o seguinte estrutura:

```

[
  {
    "{#SNMPINDEX}": 1,
    "{#IFDESCR}": "Interface #1",
    "{#IFALIAS}": "eth0"
  },
  {
    "{#SNMPINDEX}": 2,
    "{#IFDESCR}": "Interface #2",
    "{#IFALIAS}": "eth1"
  },
  {
    "{#SNMPINDEX}": 3,
    "{#IFALIAS}": "eth2"
  },
  {
    "{#SNMPINDEX}": 4,
    "{#IFDESCR}": "Interface #4"
  },
  {
    "{#SNMPINDEX}": 5,
    "{#IFALIAS}": "eth4"
  }
]

```

Protótipos de itens

A captura de tela a seguir ilustra como podemos usar essas macros no item protótipos:

Item prototype	Tags	Preprocessing 2
* Name	Interface {#IFNAME}{#IFALIAS}: Bits received	
Type	SNMP agent	
* Key	net.if.in[ifHCInOctets.{#SNMPINDEX}]	
Type of information	Numeric (unsigned)	
* SNMP OID	1.3.6.1.2.1.31.1.1.1.6.{#SNMPINDEX}	
Units	bps	
* Update interval	3m	

Novamente, criando quantos protótipos de itens forem necessários:

Item prototypes

All templates / Linux SNMP		Discovery list / Network interfaces discovery		Item prototypes 9	Trigger prototypes 4	Graph prototypes 1	Host prototypes	
<input type="checkbox"/>	Name ▲	Key	Interval	History	Trends	Type	Create enabled	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Bits received	net.if.in[ifHCInOctets.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Bits sent	net.if.out[ifHCOutOctets.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Inbound packets discarded	net.if.in.discards[ifInDiscards.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Inbound packets with errors	net.if.in.errors[ifInErrors.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Interface type	net.if.type[ifType.{#SNMPINDEX}]	1h	7d	0d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Operational status	net.if.status[ifOperStatus.{#SNMPINDEX}]	1m	7d	0	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Outbound packets discarded	net.if.out.discards[ifOutDiscards.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Outbound packets with errors	net.if.out.errors[ifOutErrors.{#SNMPINDEX}]	3m	7d	365d	SNMP agent	Yes	
<input type="checkbox"/>	... Interface {#IFNAME}{(#IFALIAS)}: Speed	net.if.speed[ifHighSpeed.{#SNMPINDEX}]	5m	7d	0d	SNMP agent	Yes	

Protótipos de gatilho

A captura de tela a seguir ilustra como podemos usar essas macros em protótipos de gatilho:

* Name Interface {#IFNAME}{#IFALIAS}: Link down

Event name Interface {#IFNAME}{#IFALIAS}: Link down

Operational data Current state: {ITEM.LASTVALUE1}

Severity

Not classified

Information

Warning

Average

High

Disaster

* Problem expression

```
{$IFCONTROL:"{#IFNAME}"}=1 and last(/SNMP
host/net.if.status[ifOperStatus.{#SNMPINDEX}])=2 and
(last(/SNMP host/net.if.status[ifOperStatus.
{#SNMPINDEX}],#1)<>last(/SNMP
host/net.if.status[ifOperStatus.{#SNMPINDEX}],#2))
```

[Expression constructor](#)

OK event generation

Expression

Recovery expression

None

* Recovery expression

```
last(/SNMP host/net.if.status[ifOperStatus.{#SNMPINDEX}])<>2
or {$IFCONTROL:"{#IFNAME}"}=0
```

Trigger prototypes

Severity	Name	Operational data	Expression	Create enabled
<input type="checkbox"/> Information	Interface {#IFNAME} {#IFALIAS}: Ethernet has changed to lower speed than it was before Depends on: Linux SNMP: Interface {#IFNAME}{#IFALIAS}: Link down	Current reported speed: {ITEM.LASTVALUE1}	Problem: <code>change(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])<0 and last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0 and (last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=6 or last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=7 or last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=11 or last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=62 or last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=69 or last(/Linux SNMP/net.if.type[ifType.{#SNMPINDEX}])=117) and (last(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])<>2)</code> Recovery: <code>(change(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0 and last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>2) or (last(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])=2)</code>	Yes
<input type="checkbox"/> Warning	Interface {#IFNAME} {#IFALIAS}: High bandwidth usage Depends on: Linux SNMP: Interface {#IFNAME}{#IFALIAS}: Link down	In: {ITEM.LASTVALUE1}, out: {ITEM.LASTVALUE3}, speed: {ITEM.LASTVALUE2}	Problem: <code>(avg(/Linux SNMP/net.if.in[ifHCInOctets.{#SNMPINDEX}],15m)>({\$IF.UTIL.MAX:"{#IFNAME}"})/100)*last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) or avg(/Linux SNMP/net.if.out[ifHCOutOctets.{#SNMPINDEX}],15m)>({\$IF.UTIL.MAX:"{#IFNAME}"})/100)*last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) and last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])>0</code> Recovery: <code>avg(/Linux SNMP/net.if.in[ifHCInOctets.{#SNMPINDEX}],15m)<(({\$IF.UTIL.MAX:"{#IFNAME}"})/100)*last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}]) and avg(/Linux SNMP/net.if.out[ifHCOutOctets.{#SNMPINDEX}],15m)<(({\$IF.UTIL.MAX:"{#IFNAME}"})/100)*last(/Linux SNMP/net.if.speed[ifHighSpeed.{#SNMPINDEX}])</code>	Yes
<input type="checkbox"/> Warning	Interface {#IFNAME} {#IFALIAS}: High error rate Depends on: Linux SNMP: Interface {#IFNAME}{#IFALIAS}: Link down	errors in: {ITEM.LASTVALUE1}, errors out: {ITEM.LASTVALUE2}	Problem: <code>min(/Linux SNMP/net.if.in.errors[ifInErrors.{#SNMPINDEX}],5m)>{\$IF.ERRORS.WARN:"{#IFNAME}"}) or min(/Linux SNMP/net.if.out.errors[ifOutErrors.{#SNMPINDEX}],5m)>{\$IF.ERRORS.WARN:"{#IFNAME}"})</code> Recovery: <code>max(/Linux SNMP/net.if.in.errors[ifInErrors.{#SNMPINDEX}],5m)<{\$IF.ERRORS.WARN:"{#IFNAME}"})*0.8 and max(/Linux SNMP/net.if.out.errors[ifOutErrors.{#SNMPINDEX}],5m)<{\$IF.ERRORS.WARN:"{#IFNAME}"})*0.8</code>	Yes
<input type="checkbox"/> Average	Interface {#IFNAME} {#IFALIAS}: Link down	Current state: {ITEM.LASTVALUE1}	Problem: <code>{\$IFCONTROL:"{#IFNAME}"}=1 and last(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}])=2 and (last(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}],#1)<>last(/Linux SNMP/net.if.status[ifOperStatus.{#SNMPINDEX}],#2))</code>	Yes

Protótipos de gráficos

A captura de tela a seguir ilustra como podemos usar essas macros em protótipos de gráficos:

Graph prototype

Preview

* Name

Interface {#IFNAME}({#IFALIAS}): Network traffic

* Width

900

* Height

200

Graph type

Normal

Show legend

☒

Show working time

☒

Show triggers

☒

Percentile line (left)

☐

Percentile line (right)

☐

Y axis MIN value

Calculated

Y axis MAX value

Calculated

* Items

	Name	Function	Draw style	Y axis side	Color
1:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Bits received	avg	Gradient line	Left	<div></div>
2:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Bits sent	avg	Bold line	Left	<div></div>
3:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Outbound packets with errors	avg	Line	Right	<div></div>
4:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Inbound packets with errors	avg	Line	Right	<div></div>
5:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Outbound packets discarded	avg	Line	Right	<div></div>
6:	SNMP host: Interface {#IFNAME}({#IFALIAS}): Inbound packets discarded	avg	Line	Right	<div></div>

[Add](#) [Add prototype](#)

Graph prototypes

All templates / Linux SNMP			Discovery list / Network interfaces discovery	Item prototypes 9	Trigger prototypes 4	Graph prototypes 1	Host prototypes
<input type="checkbox"/>	Name ▲					Width	Height
<input type="checkbox"/>	Interface {#IFNAME}({#IFALIAS}): Network traffic					900	200

Um resumo da nossa regra de descoberta:

All templates / Linux SNMP			Items 26	Triggers 10	Graphs 5	Dashboards 2	Discovery rules 5	Web scenarios
<input type="checkbox"/>	Template	Name ▲		Items		Triggers		Graphs
<input type="checkbox"/>	Linux SNMP	Network interfaces discovery		Item prototypes 9		Trigger prototypes 4		Graph prototypes 1

Entidades descobertas

Quando o servidor for executado, ele criará itens reais, gatilhos e gráficos baseados nos valores que a regra de descoberta SNMP retorna. Na configuração do host eles são prefixados com um link laranja para uma regra de descoberta eles vêm a partir de.

Items

All hosts / SNMP host Enabled SNMP Items 81 Triggers 23 Graphs 14 Discovery rules 6 Web scenarios										
<input type="checkbox"/>	Name ▲	Triggers	Key	Interval	History	Trends	Type	Status		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Bits received	Triggers 1	net.if.in[ifHCInOctets.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Bits sent	Triggers 1	net.if.out[ifHCOutOctets.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Inbound packets discarded		net.if.in.discards[ifInDiscards.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Inbound packets with errors	Triggers 1	net.if.in.errors[ifInErrors.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Interface type	Triggers 1	net.if.type[ifType.2]	1h	7d	0d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Operational status	Triggers 2	net.if.status[ifOperStatus.2]	1m	7d	0	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Outbound packets discarded		net.if.out.discards[ifOutDiscards.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Outbound packets with errors	Triggers 1	net.if.out.errors[ifOutErrors.2]	3m	7d	365d	SNMP agent	Enabled		
<input type="checkbox"/>	... Network interfaces discovery: Interface enp4s0(): Speed	Triggers 2	net.if.speed[ifHighSpeed.2]	5m	7d	0d	SNMP agent	Enabled		

Triggers

All hosts / SNMP host Enabled SNMP Items 81 Triggers 23 Graphs 14 Discovery rules 6 Web scenarios					
<input type="checkbox"/>	Severity	Value	Name ▲	Operational data	Expression
<input type="checkbox"/>	Information	OK	Network interfaces discovery: Interface enp4s0(): Ethernet has changed to lower speed than it was before Depends on: SNMP host: Interface enp4s0(): Link down	Current reported speed: {ITEM.LASTVALUE1}	Problem: change (/SNMP host/net.if.speed[ifHighSpeed.2])<0 and last (/SNMP host/net.if.speed[ifHighSpeed.2])>0 and (last (/SNMP host/net.if.type[ifType.2])=6 or last (/SNMP host/net.if.type[ifType.2])=7 or last (/SNMP host/net.if.type[ifType.2])=11 or last (/SNMP host/net.if.type[ifType.2])=62 or last (/SNMP host/net.if.type[ifType.2])=69 or last (/SNMP host/net.if.type[ifType.2])=117) and (last (/SNMP host/net.if.status[ifOperStatus.2])<>2) Recovery: (change (/SNMP host/net.if.speed[ifHighSpeed.2])>0 and last (/SNMP host/net.if.speed[ifHighSpeed.2],#2)>0) or (last (/SNMP host/net.if.status[ifOperStatus.2])=2)
<input type="checkbox"/>	Warning	OK	Network interfaces discovery: Interface enp4s0(): High bandwidth usage Depends on: SNMP host: Interface enp4s0(): Link down	In: {ITEM.LASTVALUE1}, out: {ITEM.LASTVALUE3}, speed: {ITEM.LASTVALUE2}	Problem: (avg (/SNMP host/net.if.in[ifHCInOctets.2],15m)>({\$IF.UTIL.MAX:"enp4s0"}/100)* last (/SNMP host/net.if.speed[ifHighSpeed.2]) or avg (/SNMP host/net.if.out[ifHCOutOctets.2],15m)>({\$IF.UTIL.MAX:"enp4s0"}/100)* last (/SNMP host/net.if.speed[ifHighSpeed.2])) and last (/SNMP host/net.if.speed[ifHighSpeed.2])>0 Recovery: avg (/SNMP host/net.if.in[ifHCInOctets.2],15m)<(({\$IF.UTIL.MAX:"enp4s0"}-3)/100)* last (/SNMP host/net.if.speed[ifHighSpeed.2]) and avg (/SNMP host/net.if.out[ifHCOutOctets.2],15m)<(({\$IF.UTIL.MAX:"enp4s0"}-3)/100)* last (/SNMP host/net.if.speed[ifHighSpeed.2])
<input type="checkbox"/>	Warning	OK	Network interfaces discovery: Interface enp4s0(): High error rate Depends on: SNMP host: Interface enp4s0(): Link down	errors in: {ITEM.LASTVALUE1}, errors out: {ITEM.LASTVALUE2}	Problem: min (/SNMP host/net.if.in.errors[ifInErrors.2],5m)>{\$IF.ERRORS.WARN:"enp4s0"} or min (/SNMP host/net.if.out.errors[ifOutErrors.2],5m)>{\$IF.ERRORS.WARN:"enp4s0"} Recovery: max (/SNMP host/net.if.in.errors[ifInErrors.2],5m)<{\$IF.ERRORS.WARN:"enp4s0"}*0.8 and max (/SNMP host/net.if.out.errors[ifOutErrors.2],5m)<{\$IF.ERRORS.WARN:"enp4s0"}*0.8
<input type="checkbox"/>	Average	OK	Network interfaces discovery: Interface enp4s0(): Link down	Current state: {ITEM.LASTVALUE1}	Problem: {\$IFCONTROL:"enp4s0"}=1 and last (/SNMP host/net.if.status[ifOperStatus.2])=2 and (last (/SNMP host/net.if.status[ifOperStatus.2],#1)<< last (/SNMP host/net.if.status[ifOperStatus.2],#2)) Recovery: last (/SNMP host/net.if.status[ifOperStatus.2])<>2 or {\$IFCONTROL:"enp4s0"}=0

≡ Graphs

All hosts / SNMP host
Enabled
SNMP
Items 81
Triggers 23
Graphs 14
Discovery rules 6
Web scenarios

☐ Name ▲

☐ Mounted filesystem discovery: /: Disk space usage

☐ Linux SNMP: CPU jumps

☐ CPU discovery: CPU usage

☐ CPU discovery: CPU utilization

☐ Network interfaces discovery: Interface enp4s0(): Network traffic

5 Descoberta de objetos JMX

Visão geral

É possível **descobrir** todos JMX MBeans ou atributos MBean ou para especificar um padrão para a descoberta desses objetos.

É obrigatório entender a diferença entre um MBean e um MBean atributos para configuração de regras de descoberta. Um MBean é um objeto que pode representar um dispositivo, um aplicativo ou qualquer recurso que precise ser gerenciado.

Por exemplo, existe um MBean que representa um servidor web. Seus atributos são contagem de conexão, contagem de threads, tempo limite de solicitação, http cache de arquivos, uso de memória, etc. linguagem abrangente podemos definir uma máquina de café como um MBean que possui os seguintes atributos a serem monitorados: quantidade de água por xícara, consumo médio de água por um determinado período de tempo, número de grãos de café necessários por xícara, grãos de café e tempo de recarga de água, etc.

Chave do item

Em **descoberta regra** configuração, selecione **Agente JMX** no campo Tipo.

Duas chaves de item são suportadas para descoberta de objeto JMX - jmx.discovery[] e jmx.get[]:

Chave do item

	Valor de retorno	Parâmetros	Comentário
jmx.discovery [<modo de descoberta>,<nome do objeto>,<descrição curta única>]			

Este item retorna um array JSON com macros LLD de-screvendo objetos MBean ou seus atributos.	modo de descoberta - um dos seguintes: at-tributes (recu-perar atributos MBean JMX, padrão) ou beans (recu-perar JMX MBeans) nome do objeto - padrão de nome do objeto (consulte documentação) identificando o Nomes MBean a serem recuperados (vazios por padrão, recuperando todos os beans registristados) descrição curta única - uma descrição exclusiva que	Exemplos: : → jmx.discovery - recu-pera todos os atributos JMX MBean → jmx.discovery[beans] - recu-pera todos os JMX MBeans → jmx.discovery[attribute name=*"] - recu-pera todos os atributos do coletor de lixo → jmx.discovery[beans, - recu-pera todos os coletores de lixo Existem algumas limitações ao que Propriedades MBean este item pode re-tornar com base em caracte-res limita-dos que são supor-tados na geração
--	---	--

Chave do item

jmx.get[<modo de descoberta>,<nome do objeto>,<descrição curta única>]

Este item retorna um array JSON com objetos MBean ou seus atributos. Comparado com `jmx.discovers` ele não define macros LLD.

modo de descoberta

- um dos seguintes atributos (recuperar atributos JMX MBean, padrão) ou `beans []`, (recuperar valores extraídos dos do JSON retornado usando JSON-Path.

nome do objeto (consulte [documentação](#)) identificando os nomes MBean a serem recuperados (vazios por padrão, recuperando todos os beans registrados)

descrição curta única - uma descrição única que permite

Ao usar este item, é necessário definir macros de descoberta de baixo nível personalizado,izadas, adicionando, para valores extraídos dos do JSON

br>Suportado desde o Zabbix Java gateway 4.4.

Chave do item

::: não importante Se nenhum parâmetro for passado, todos os atributos MBean do JMX são solicitados. Não especificando parâmetros para descoberta JMX ou tentando receber todos os atributos para um amplo intervalo como *:type=*,name=* pode levar a potenciais problemas de desempenho. :::

Usando jmx.discovery

Este item retorna um objeto JSON com macros de descoberta de baixo nível descrevendo objetos ou atributos MBean. Por exemplo, na descoberta de Atributos MBean (reformatados para maior clareza):

```
[
  {
    "#{JMXVALUE}": "0",
    "#{JMXTYPE}": "java.lang.Long",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXDESC}": "java.lang:type=GarbageCollector,name=PS Scavenge,CollectionCount",
    "#{JMXATTR}": "Contagem de Coleção"
  },
  {
    "#{JMXVALUE}": "0",
    "#{JMXTYPE}": "java.lang.Long",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXDESC}": "java.lang:type=GarbageCollector,name=PS Scavenge,CollectionTime",
    "#{JMXATTR}": "Hora da Coleção"
  },
  {
    "#{JMXVALUE}": "true",
    "#{JMXTYPE}": "java.lang.Boolean",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXDESC}": "java.lang:type=GarbageCollector,name=PS Scavenge,Valid",
    "#{JMXATTR}": "Válido"
  },
  {
    "#{JMXVALUE}": "Limpeza PS",
    "#{JMXTYPE}": "java.lang.String",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXDESC}": "java.lang:type=GarbageCollector,name=PS Scavenge,Name",
    "#{JMXATTR}": "Nome"
  },
  {
    "#{JMXVALUE}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXTYPE}": "javax.management.ObjectName",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXDESC}": "java.lang:type=GarbageCollector,name=PS Scavenge,ObjectName",
    "#{JMXATTR}": "Nome do Objeto"
  }
]
```

Na descoberta de MBeans (reformatado para maior clareza):

```
[
  {
    "#{JMXDOMAIN}": "java.lang",
    "#{JMXTYPE}": "Coletor de Lixo",
    "#{JMXOBJ}": "java.lang:type=GarbageCollector,name=PS Scavenge",
    "#{JMXNAME}": "Limpeza PS"
  }
]
```

Macros compatíveis

As seguintes macros têm suporte para uso na regra de descoberta **filtro** e protótipos de itens, gatilhos e gráficos:

Macro	Descrição
Descoberta de atributos MBean	
{#JMXVALUE}	Valor do atributo.
{#JMXTYPE}	Tipo de atributo.
{#JMXOBJ}	Nome do objeto.
{#JMXDESC}	Nome do objeto incluindo o nome do atributo.
{#JMXATTR}	Nome do atributo.
Descoberta de MBeans	
{#JMXDOMAIN}	domínio MBean. (Nome reservado Zabbix)
{#JMXOBJ}	Nome do objeto. (Nome reservado Zabbix)
{#JMX<propriedade da chave>}	Propriedades MBean (como {#JMXTYPE}, {#JMXNAME}) (consulte Limitações abaixo).

Limitações

Existem algumas limitações associadas ao algoritmo de criação de LLD nomes de macro de nomes de propriedade MBean:

- os nomes dos atributos são alterados para maiúsculas
- nomes de atributos são ignorados (nenhuma macro LLD é gerada) se eles consistem em caracteres não suportados para nomes de macro LLD. Compatível caracteres podem ser descritos pela seguinte expressão regular: A-Z0-9_\. .
- se um atributo for chamado de "obj" ou "domínio" ele será ignorado por causa da sobreposição com os valores do Zabbix reservado propriedades {#JMXOBJ} e {#JMXDOMAIN} (suportado desde o Zabbix 3.4.3.)

Por favor, considere este exemplo jmx.discovery (com modo "beans"). MBean tem as seguintes propriedades definidas:

```
nome=teste
=Tipo
atributos []=1,2,3
Nome=NomeDoTeste
domínio=algum
```

Como resultado da descoberta JMX, as seguintes macros LLD serão gerado:

- {#JMXDOMAIN} - Zabbix interno, descrevendo o domínio do MBean
- {#JMXOBJ} - Zabbix interno, descrevendo o objeto MBean
- {#JMXNAME} - criado a partir da propriedade "name"

As propriedades ignoradas são:

- тип : seu nome contém caracteres não suportados (não ASCII)
- attribute[] : seu nome contém caracteres não suportados (quadrado colchetes não são suportados)
- Nome: já está definido (nome=teste)
- domAin : é um nome reservado do Zabbix

Exemplos

Vamos revisar mais dois exemplos práticos de criação de uma regra LLD com o uso do Mbean. Para entender a diferença entre a coleta de uma regra LLD Mbeans e uma regra LLD coletando melhor os atributos Mbean, por favor, dê uma olhada veja a tabela a seguir:

MBean1	MBean2	MBean3
MBean1Attribute1	MBean2Attribute1	MBean3Attribute1
MBean1Attribute2	MBean2Attribute2	MBean3Attribute2
MBean1Attribute3	MBean2Attribute3	MBean3Attribute3

Exemplo 1: Descobrindo Mbeans

Esta regra retornará 3 objetos: a linha superior da coluna: MBean1, MBean2, MBean3.

Para obter mais informações sobre objetos, consulte [suportado macros](#) tabela, seção Descoberta de MBeans.

Configuração da regra de descoberta coletando Mbeans (sem os atributos) se parece com o seguinte:

Discovery rule
Preprocessing
LLD macros
Filters
Overrides

* Name
JMX garbage collectors

Type
JMX agent

* Key
jmx.discovery[beans,"*:type=GarbageCollector,name=*"]

* Host interface
127.0.0.1 : 12345

A chave usada aqui:

```
jmx.discovery[beans,"*:type=GarbageCollector,name=*"]
```

Todos os coletores de lixo sem atributos serão descobertos. Como Os coletores de lixo têm o mesmo conjunto de atributos, podemos usar atributos em protótipos de itens da seguinte maneira:

Item prototypes

All hosts / JMX
Enabled
JMX
Discovery list / JMX garbage collectors
Item prototypes
Trigger p

<input type="checkbox"/> Name ▲	Key
<input type="checkbox"/> GC {#JMXNAME} CollectionCount	jmx[{#JMXOBJ},CollectionCount]
<input type="checkbox"/> GC {#JMXNAME} CollectionTime	jmx[{#JMXOBJ},CollectionTime]
<input type="checkbox"/> GC {#JMXNAME} Valid	jmx[{#JMXOBJ},Valid]

As chaves usadas aqui:

```
jmx[{#JMXOBJ},CollectionCount]
jmx[{#JMXOBJ},Hora da Coleção]
jmx[{#JMXOBJ},Válido]
```

A regra de descoberta de LLD resultará em algo próximo a isso (os itens são descoberto por dois coletores de lixo):

	Triggers	Key
<input type="checkbox"/> ... JMX garbage collectors: GC PS MarkSweep CollectionCount		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",CollectionCount]
<input type="checkbox"/> ... JMX garbage collectors: GC PS MarkSweep CollectionTime		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",CollectionTime]
<input type="checkbox"/> ... JMX garbage collectors: GC PS MarkSweep Valid		jmx["java.lang:type=GarbageCollector,name=PS MarkSweep",Valid]
<input type="checkbox"/> ... JMX garbage collectors: GC PS Scavenge CollectionCount		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",CollectionCount]
<input type="checkbox"/> ... JMX garbage collectors: GC PS Scavenge CollectionTime		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",CollectionTime]
<input type="checkbox"/> ... JMX garbage collectors: GC PS Scavenge Valid		jmx["java.lang:type=GarbageCollector,name=PS Scavenge",Valid]

Exemplo 2: Descobrendo atributos do Mbean

Esta regra retornará 9 objetos com os seguintes campos: MBean1Atributo1, MBean2Atributo1, Mbean3Attribute1,MBean1Attribute2,MBean2Atributo2, MBean3Attribute2, MBean1Attribute3, MBean2Attribute3, Mbean3Attribute3.

Para obter mais informações sobre objetos, consulte [suportado macros](#) tabela, seção Descoberta de atributos MBean.

A configuração da regra de descoberta coletando atributos Mbean se parece com o Segue:

Discovery rule	Preprocessing	LLD macros	Filters	Overrides
* Name	JMX garbage collectors			
Type	JMX agent			
* Key	jmx.discovery[attributes,"*:type=GarbageCollector,name=*"]			
* Host interface	127.0.0.1 : 12345			

A chave usada aqui:

```
jmx.discovery[attributes,"*:type=GarbageCollector,name=*"]
```

Todos os coletores de lixo com um único atributo de item serão descoberto.

Item prototypes

All hosts / JMX	Enabled	JMX	Discovery list / JMX garbage collectors	Item prototypes	Trigger
<input type="checkbox"/> Name ▲	Key				
<input type="checkbox"/> {#JMXOBJ} {#JMXATTR}	jmx[{#JMXOBJ},{#JMXATTR}]				

Neste caso em particular, um item será criado a partir do protótipo para cada Atributo MBean. A principal desvantagem dessa configuração é que acionar a criação de protótipos de gatilho é impossível, pois há apenas um item protótipo para todos os atributos. Portanto, esta configuração pode ser usada para dados coleta, mas não é recomendado para monitoramento automático.

Usando `jmx.get`

`jmx.get []` é semelhante ao item `jmx.discovery []`, mas não é transformar propriedades de objetos Java em nomes de macros de descoberta de baixo nível e portanto, pode retornar valores sem **limitações** que são associado à geração de nome de macro LLD, como hífen ou não ASCII personagens.

Ao usar `jmx.get []` para descoberta, macros de descoberta de baixo nível podem ser definido separadamente no personalizado **LLD macro** da guia configuração de regra de descoberta, usando JSONPath para apontar para o valores.

Descobrendo MBeans

Item de descoberta: `jmx.get[beans,"com.example:type=*,*"]`

Resposta:

```
[
  {
    "object": "com.example:type=Hello,data-src=data-base, = ",
    "domínio": "com.exemplo",
    "propriedades": {
      "data-src": "base de dados",
      " ": " ",
      "tipo": "Olá"
    }
  },
  {
    "object": "com.example:type=Atomic",
    "domínio": "com.exemplo",
    "propriedades": {
      "tipo": "Atômica"
    }
  }
]
```

```
}  
]
```

Descobrimos os atributos do MBean

Item de descoberta: `jmx.get[attributes,"com.example:type=*,*"]`

Resposta:

```
[  
  {  
    "object": "com.example:type=*",  
    "domínio": "com.exemplo",  
    "propriedades": {  
      "tipo": "Simples"  
    }  
  },  
  {  
    "object": "com.zabbix:type=yes, domain=zabbix.com, data-source=/dev/rand, = ,obj=true",  
    "domínio": "com.zabbix",  
    "propriedades": {  
      "tipo": "Olá",  
      "domínio": "com.exemplo",  
      "fonte de dados": "/dev/rand",  
      " ": " ",  
      "obj": verdadeiro  
    }  
  }  
]
```

6 Descoberta de sensores IPMI

Visão geral

É possível descobrir automaticamente os sensores IPMI.

Para fazer isso, você pode usar uma combinação de:

- o item IPMI `ipmi.get` (suportado desde o Zabbix **5.0.0**) como o item mestre
- regra de descoberta de baixo nível dependente e protótipos de itens

1 Configuração

Item mestre

Crie um item IPMI usando a seguinte chave:

`ipmi.get`

Item	Tags	Preprocessing
<div>* Name <input type="text" value="IPMI get item"/></div>		
<div>Type <input type="text" value="IPMI agent"/></div>		
<div>* Key <input type="text" value="ipmi.get"/></div>		
<div>* Host interface <input type="text" value="127.0.0.1 : 623"/></div>		
<div>IPMI sensor <input type="text"/></div>		
<div>Type of information <input type="text" value="Text"/></div>		

Defina o tipo de informação como "Texto" para dados JSON possivelmente grandes.

Regra LLD dependente

Crie uma regra de descoberta de baixo nível como tipo "Item dependente":

Discovery rule	Preprocessing	LLD macros	Filters	Overrides
<div>* Name <input type="text" value="Discovery rule for ipmi.get"/></div>				
<div>Type <input type="text" value="Dependent item"/></div>				
<div>* Key <input type="text" value="ipmi.sensor.discovery"/></div>				
<div>* Master item <input type="text" value="Zabbix server: IPMI get item"/></div>				

Como item mestre selecione o item `ipmi.get` que criamos.

Na aba "Macros LLD" defina uma macro personalizada com o correspondente Caminho JSON:

Discovery rule	Preprocessing	LLD macros 1	Filters	Overrides				
<div>LLD macros</div> <table border="1"> <thead> <tr> <th>LLD macro</th> <th>JSONPath</th> </tr> </thead> <tbody> <tr> <td><input data-bbox="517 1771 1182 1834" type="text" value="{#SENSOR_ID}"/></td> <td><input data-bbox="1198 1771 1453 1834" type="text" value="\$..id"/></td> </tr> </tbody> </table> <div>Add</div>					LLD macro	JSONPath	<input data-bbox="517 1771 1182 1834" type="text" value="{#SENSOR_ID}"/>	<input data-bbox="1198 1771 1453 1834" type="text" value="\$..id"/>
LLD macro	JSONPath							
<input data-bbox="517 1771 1182 1834" type="text" value="{#SENSOR_ID}"/>	<input data-bbox="1198 1771 1453 1834" type="text" value="\$..id"/>							

Protótipo de item dependente

Crie um protótipo de item com o tipo "Item dependente" nesta regra LLD. Como item mestre para este protótipo selecione o item `ipmi.get` que criamos.

Item prototype	Tags	Preprocessing
<div> <div>* Name</div> <div>IPMI value for sensor {#SENSOR_ID}</div> </div>		
<div> <div>Type</div> <div>Dependent item</div> </div>		
<div> <div>* Key</div> <div>ipmi_sensor[{#SENSOR_ID}]</div> </div>		
<div> <div>* Master item</div> <div>Zabbix server: IPMI get item</div> </div>		
<div> <div>Type of information</div> <div>Numeric (unsigned)</div> </div>		

Observe o uso da macro {#SENSOR_ID} no nome do protótipo do item e chave:

- Nome: valor IPMI para o sensor {#SENSOR_ID}
- Chave: ipmi_sensor[{#SENSOR_ID}]

Como tipo de informação, Numérico (sem sinal).

Na guia "Pré-processamento" do protótipo do item, selecione JSONPath e use o seguinte expressão JSONPath como parâmetro:

`$. [?(@.id=='{#SENSOR_ID}')].value.first()`

Item prototype	Tags	Preprocessing 1
<div> <div>Preprocessing steps</div> <div>Name</div> <div>Parameters</div> </div>		
<div> <div>1:</div> <div>JSONPath</div> <div>\$. [?(@.id=='{#SENSOR_ID}')].value.first()</div> </div>		
<div>Add</div>		

Quando a descoberta for iniciada, será criado um item para cada sensor IPMI. Este item retornará o valor inteiro do sensor fornecido.

7 Descoberta de serviços systemd

Visão geral

É possível **descobrir** unidades systemd (serviços, por padrão) com Zabbix.

Chave do item

O item a ser usado na **descoberta regra** é a

`systemd.unit.discovery`

::: não importante este **item** é suportado apenas no agente Zabbix 2. :::

Este item retorna um JSON com informações sobre unidades systemd, por exemplo:

```
[{
  "{#UNIT.NAME}": "mysqld.service",
  "{#UNIT.DESCRPTION}": "Servidor MySQL",
  "{#UNIT.LOADSTATE}": "carregado",
  "{#UNIT.ACTIVESTATE}": "ativo",
  "{#UNIT.SUBSTATE}": "executando",
  "{#UNIT.FOLLOWED}": "",
  "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/mysqld_2eservice",
  "{#UNIT.JOBID}": 0,
  "{#UNIT.JOBTYP}": ""
}]
```

```

    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "habilitado"
  }, {
    "{#UNIT.NAME}": "systemd-journald.socket",
    "{#UNIT.DESCRPTION}": "Soquete de diário",
    "{#UNIT.LOADSTATE}": "carregado",
    "{#UNIT.ACTIVESTATE}": "ativo",
    "{#UNIT.SUBSTATE}": "executando",
    "{#UNIT.FOLLOWED}": "",
    "{#UNIT.PATH}": "/org/freedesktop/systemd1/unit/systemd_2djournald_2esocket",
    "{#UNIT.JOBID}": 0,
    "{#UNIT.JOBTYPE}": "",
    "{#UNIT.JOBPATH}": "/",
    "{#UNIT.UNITFILESTATE}": "habilitado"
  }]

```

Descoberta de unidades systemd desabilitadas

Desde o Zabbix 6.0.1 também é possível descobrir unidades systemd **desativadas**. Nesse caso, três macros são retornadas no JSON resultante:

- {#UNIT.PATH}
- {#UNIT.ACTIVESTATE}
- {#UNIT.UNITFILESTATE}.

Attention:

Para ter itens e acionadores criados a partir de protótipos para unidades do systemd desabilitadas, certifique-se de ajustar (ou remover) filtros de proibição de LLD para {#UNIT.ACTIVESTATE} e {#UNIT.UNITFILESTATE}.

Macros compatíveis

As seguintes macros têm suporte para uso na regra de descoberta **filtro** e protótipos de itens, gatilhos e gráficos:

Macro	Descrição
{#UNIT.NAME}	Nome da unidade primária.
{#UNIT.DESCRPTION}	Descrição legível por humanos.
{#UNIT.LOADSTATE}	Estado de carregamento (ou seja, se o arquivo de unidade foi carregado com sucesso)
{#UNIT.ACTIVESTATE}	Estado ativo (ou seja, se a unidade está iniciada ou não)
{#UNIT.SUBSTATE}	Subestado (uma versão mais refinada do estado ativo que é específico para o tipo de unidade, que o estado ativo não é)
{#UNIT.FOLLOWED}	Unidade que está sendo seguida em seu estado por esta unidade, se houver; caso contrário, uma string vazia.
{#UNIT.PATH}	Caminho do objeto da unidade.
{#UNIT.JOBID}	ID de trabalho numérico se houver um trabalho na fila para a unidade de trabalho; 0 caso contrário.
{#UNIT.JOBTYPE}	Tipo de trabalho.
{#UNIT.JOBPATH}	Caminho do objeto de trabalho.
{#UNIT.UNITFILESTATE}	O estado de instalação do arquivo de unidade.

Protótipos de itens

Protótipos de itens que podem ser criados com base na descoberta de serviços do systemd incluem, por exemplo:

- Nome do item: {#UNIT.DESCRPTION}; chave do item: `systemd.unit.info["{#UNIT.NAME}"]`
- Nome do item: {#UNIT.DESCRPTION}; chave do item: `systemd.unit.info["{#UNIT.NAME}",LoadState]`

`systemd.unit.info` **agente items** são suportado desde o Zabbix 4.4.

8 Descoberta de serviços do Windows

Visão geral

De maneira semelhante ao **arquivo sistemas** são descobertos, é possível descobrir também os serviços do Windows.

Chave do item

O item a ser usado na **descoberta regra** é

`serviço.descoberta`

Este item é suportado desde o Zabbix Windows agent 3.0.

Macros compatíveis

As seguintes macros têm suporte para uso na regra de descoberta **filtro** e protótipos de itens, gatilhos e gráficos:

Macro	Descrição
<code>{#SERVICE.NAME}</code>	Nome do serviço.
<code>{#SERVICE.DISPLAYNAME}</code>	Nome do serviço exibido.
<code>{#SERVICE.DESCRPTION}</code>	Descrição do serviço.
<code>{#SERVICE.STATE}</code>	Valor numérico do estado do serviço: 0 - Em execução 1 - Pausado 2 - Início pendente 3 - Pausa pendente 4 - Continuar pendente 5 - Parada pendente 6 - Parada 7 - Desconhecido
<code>{#SERVICE.STATENAME}</code>	Nome do estado do serviço (Em execução, Pausado, Início pendente, Pausa pendente, Continuar pendente, Parar pendente, Parado ou Desconhecido).
<code>{#SERVICE.PATH}</code>	Caminho do serviço.
<code>{#SERVICE.USER}</code>	Usuário do serviço.
<code>{#SERVICE.STARTUP}</code>	Valor numérico do tipo de inicialização do serviço: 0 - Automático 1 - Automático atrasado 2 - Manual 3 - Desativado 4 - Desconhecido
<code>{#SERVICE.STARTUPNAME}</code>	Nome do tipo de inicialização do serviço (Automático, Automático atrasado, Manual, Desativado, Desconhecido).
<code>{#SERVICE.STARTUPTRIGGER}</code>	Valor numérico para indicar se o tipo de inicialização do serviço tem: 0 - sem gatilhos de inicialização 1 - tem gatilhos de inicialização Esta macro é suportada desde o Zabbix 3.4.4. É útil descobrir tipos de inicialização de serviço como Automático (inicialização do acionador), Atrasado automático (inicialização do acionador) e Manual (inicialização do acionador).

Com base na descoberta de serviços do Windows, você pode criar um **item** protótipo Como

`service.info[{#SERVICE.NAME},<param>]`

onde `param` aceita os seguintes valores: `state`, `displayname`, `caminho`, `usuário`, `inicialização` ou `descrição`.

Por exemplo, para adquirir o nome de exibição de um serviço, você pode usar um `"service.info[{#SERVICE.NAME},displayname]"` item. Se o valor `param` for não especificado (`"service.info[{#SERVICE.NAME}]"`), o estado padrão parâmetro é usado.

9 Descoberta de instâncias do contador de desempenho do Windows

Visão geral

É possível **discover** instâncias de objeto dos contadores de desempenho do Windows. Isso é útil para contadores de desempenho de várias instâncias.

Chave do item

O item a ser usado na **descoberta regra** é

`perf_instance.discovery[objeto]`

ou, para poder fornecer o nome do objeto apenas em inglês, independentemente de localização do SO:

```
perf_instance_en.discovery[objeto]
```

Por exemplo:

```
perf_instance.discovery[Processador]
perf_instance_en.discovery[Processador]
```

Esses itens são suportados desde o Zabbix Windows agent 5.0.1.

Macros compatíveis

A descoberta retornará todas as instâncias do objeto especificado no macro `{#INSTANCE}`, que pode ser usada nos protótipos de `perf_count` e `perf_count_en` itens.

```
[
    {"{#INSTANCE}": "0"},
    {"{#INSTANCE}": "1"},
    {"{#INSTANCE}": "_Total"}
]
```

Por exemplo, se a chave do item usada na regra de descoberta for:

```
perf_instance.discovery[Processador]
```

you pode criar um protótipo de item:

```
perf_counter["\\Processor({#INSTANCE})\\% Tempo do Processador"]
```

Notas:

- Se o objeto especificado não for encontrado ou não suportar variável instâncias, o item de descoberta se tornará NÃO SUPORTADO.
- Se o objeto especificado suporta instâncias de variáveis, mas atualmente não tiver nenhuma instância, uma matriz JSON vazia será retornou.
- No caso de instâncias duplicadas, elas serão ignoradas.

10 Descoberta usando consultas WMI

Visão geral

WMI é uma interface poderosa no Windows que pode ser usada para recuperar várias informações sobre componentes do Windows, serviços, estado e software instalado.

Ele pode ser usado para descoberta de disco físico e seus dados de desempenho coleção, descoberta de interface de rede, descoberta de convidado Hyper-V, monitorando os serviços do Windows e muitas outras coisas no sistema operacional Windows.

Este tipo de baixo nível **discovery** é feito usando WQL consultas cujos resultados são transformados automaticamente em um objeto JSON adequado para descoberta de baixo nível.

Chave do item

O item a ser usado na **descoberta regra** é

```
wmi.getall[<namespace>,<consulta>]
```

Este **item** transforma o resultado da consulta em uma matriz JSON. Por exemplo:

selecione * de Win32_DiskDrive onde Nome como '%PHYSICALDRIVE%'

pode retornar algo assim:

```
[
  {
    "DeviceID" : "\\.\PHYSICALDRIVE0",
    "BytesPorSetor" : 512,
    "Recursos" : [
      3,
      4
    ],
    "Descrições de Capacidade" : [
      "Acesso aleatório",
      "Suporta Escrita"
    ],
  },
]
```

```

    "Caption": "Dispositivo VBOX HARDDISK ATA",
    "ConfigManagerErrorCode" : "0",
    "ConfigManagerUserConfig": "false",
    "CreationClassName": "Win32_DiskDrive",
    "Descrição" : "Unidade de disco",
    "FirmwareRevision" : "1.0",
    "Índice": 0,
    "InterfaceType": "IDE"
  },
  {
    "DeviceID" : "\\.\PHYSICALDRIVE1",
    "BytesPorSetor" : 512,
    "Recursos" : [
      3,
      4
    ],
    "Descrições de Capacidade" : [
      "Acesso aleatório",
      "Suporta Escrita"
    ],
    "Caption": "Dispositivo VBOX HARDDISK ATA",
    "ConfigManagerErrorCode" : "0",
    "ConfigManagerUserConfig": "false",
    "CreationClassName": "Win32_DiskDrive",
    "Descrição" : "Unidade de disco",
    "FirmwareRevision" : "1.0",
    "Índice" : 1,
    "InterfaceType": "IDE"
  }
]

```

Este item é suportado desde o Zabbix Windows agent 4.4.

Macros de descoberta de baixo nível

Mesmo que nenhuma macro de descoberta de baixo nível seja criada no retorno JSON, essas macros podem ser definidas pelo usuário como uma etapa adicional, usando o **LDL personalizado macro** funcionalidade com JSONPath apontando para os valores descobertos no retorno JSON.

As macros podem ser usadas para criar protótipos de itens, gatilhos, etc.

11 Descoberta usando consultas ODBC SQL

Visão geral

Este tipo de baixo nível **discovery** é feito usando SQL consultas, cujos resultados são transformados automaticamente em um objeto JSON adequado para descoberta de baixo nível.

Chave do item

As consultas SQL são executadas usando um tipo de item "Monitor de banco de dados". Portanto, a maioria das instruções sobre **ODBC monitoramento** se aplicam em para obter uma regra de descoberta "Monitor de banco de dados" funcional.

Duas chaves de item podem ser usadas nas regras de descoberta do "Monitor de banco de dados":

- **db.odbc.discovery**[<curto exclusivo descrição>,<dsn>,<string de conexão>] - este item transforma o resultado da consulta SQL em um array JSON, tornando o nomes de coluna do resultado da consulta na macro de descoberta de baixo nível nomes emparelhados com os valores de campo descobertos. Essas macros podem ser usado na criação de protótipos de itens, gatilhos, etc. Veja também: **Usando db.odbc.discovery**.
- **db.odbc.get**[<curto exclusivo descrição>,<dsn>,<string de conexão>] - este item transforma o resultado da consulta SQL em um array JSON, mantendo a nomes de colunas originais do resultado da consulta como um nome de campo em JSON emparelhado com os valores descobertos. Comparado com `db.odbc.discovery[]`, este item não cria uma descoberta de baixo nível macros no JSON retornado, portanto, não há necessidade de verificar se os nomes das colunas podem ser nomes de macros válidos. A descoberta de baixo nível macros podem ser definidas como uma etapa adicional conforme necessário, usando o **LD personalizado macro** funcionalidade com JSONPath apontando para os valores descobertos no retorno JSON. Veja também: **Usando db.odbc.get**.

Usando db.odbc.discovery

Como exemplo prático para ilustrar como a consulta SQL é transformada em JSON, vamos considerar a descoberta de baixo nível de proxies Zabbix por realizando uma consulta ODBC no banco de dados Zabbix. Isso é útil para criação automática de "zabbix[proxy,<name>,lastaccess]" **itens internos** para monitorar quais proxies estão vivos.

Vamos começar com a configuração da regra de descoberta:

Discovery rule	Preprocessing	LLD macros	Filters	Overrides
<div><div>* Name</div><div>Proxy discovery</div></div> <div><div>Type</div><div>Database monitor</div></div> <div><div>* Key</div><div>db.odbc.discovery[proxies,{SDSN}]</div></div> <div><div>User name</div><div></div></div> <div><div>Password</div><div></div></div> <div><div>* SQL query</div><div>SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid WHERE h1.status IN (5, 6) GROUP BY h1.host;</div></div> <div><div>* Update interval</div><div>30s</div></div>				

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Aqui, a seguinte consulta direta no banco de dados Zabbix é usada para selecionar todos os proxies Zabbix, juntamente com o número de hosts que eles são monitoramento. O número de hosts pode ser usado, por exemplo, para filtrar proxies vazios:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid WHERE h1.status IN (5, 6) GROUP BY h1.host;
```

anfitrião	contar
Japão 1	5
Japão 2	12
Letônia	3

3 linhas em conjunto (0,01 seg)

Pelo funcionamento interno do item "db.odbc.discovery[,{\$SDSN}]", o resultado desta consulta é transformado automaticamente no seguinte JSON:

```
[
  {
    "#HOST": "Japão 1",
    "#COUNT": "5"
  },
  {
    "#HOST": "Japão 2",
    "#COUNT": "12"
  },
  {
    "#HOST": "Letônia",
    "#COUNT": "3"
  }
]
```

```
}
]
```

Pode-se ver que os nomes das colunas se tornam nomes de macros e linhas selecionadas tornam-se os valores dessas macros.

Note:

Se não for óbvio como um nome de coluna seria transformado em um nome de macro, sugere-se usar aliases de coluna como "COUNT(h2.host) AS count" no exemplo acima.

Caso um nome de coluna não possa ser convertido em um nome de macro válido, o a regra de descoberta não é suportada, com a mensagem de erro detalhando o número da coluna incorreta. Se for desejada ajuda adicional, o valor obtido nomes de colunas são fornecidos em DebugLevel=4 no arquivo de log do servidor Zabbix:

```
$ grep db.odbc.discovery /tmp/zabbix_server.log
...
23876:20150114:153410.856 Na consulta db_odbc_discovery():'SELECT h1.host, COUNT(h2.host) FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid'
23876:20150114:153410.860 db_odbc_discovery() coluna[1]: 'host'
23876:20150114:153410.860 db_odbc_discovery() coluna[2]: 'COUNT(h2.host)'
23876:20150114:153410.860 Fim de db_odbc_discovery():NOTSUPPORTED
23876:20150114:153410.860 Item [Zabbix server:db.odbc.discovery[proxies,{ $DSN}]] erro: Não é possível interpretar a consulta SQL
```

Agora que entendemos como uma consulta SQL é transformada em um JSON objeto, podemos usar a macro {#HOST} em protótipos de itens:

Item prototype	Tags	Preprocessing
* Name	Last access time of proxy {#HOST}	
Type	Zabbix internal	
* Key	zabbix[proxy,{#HOST},lastaccess]	
Type of information	Numeric (unsigned)	
Units	unixtime	
* Update interval	60s	

Depois que a descoberta for realizada, um item será criado para cada proxy:

<input type="checkbox"/>	Name	Triggers	Key ▲
<input type="checkbox"/>	... Proxy discovery: Last access time of proxy Japan1		zabbix[proxy,Japan1,lastaccess]
<input type="checkbox"/>	... Proxy discovery: Last access time of proxy Japan2		zabbix[proxy,Japan2,lastaccess]
<input type="checkbox"/>	... Proxy discovery: Last access time of proxy Latvia		zabbix[proxy,Latvia,lastaccess]

Usando db.odbc.get

Usando db.odbc.get[,{\$DSN}] e o seguinte exemplo SQL:

```
mysql> SELECT h1.host, COUNT(h2.host) AS count FROM hosts h1 LEFT JOIN hosts h2 ON h1.hostid = h2.proxy_hostid
+-----+-----+
| anfitrião | contar |
+-----+-----+
| Japão 1 | 5 |
| Japão 2 | 12 |
| Letônia | 3 |
+-----+-----+
```


3 linhas em conjunto (0,01 seg)

este JSON será retornado:

```
[
  {
    "host": "Japão 1",
    "contar": "5"
  },
  {
    "host": "Japão 2",
    "contar": "12"
  },
  {
    "host": "Letônia",
    "contar": "3"
  }
]
```

Como você pode ver, não há macros de descoberta de baixo nível. No entanto, macros de descoberta de baixo nível personalizadas podem ser criadas no **LLD macros** de uma regra de descoberta usando JSONPath, por exemplo:

{#HOST} → \$.host

Agora esta macro {#HOST} pode ser usada em protótipos de itens:

Item prototype	Tags	Preprocessing
<div><div>*</div><div>Name</div><div>Last access time of proxy {#HOST}</div></div>		
<div><div>Type</div><div>Zabbix internal</div><div>▼</div></div>		
<div><div>*</div><div>Key</div><div>zabbix[proxy,{#HOST},lastaccess]</div></div>		
<div><div>Type of information</div><div>Numeric (unsigned)</div><div>▼</div></div>		
<div><div>Units</div><div>unixtime</div></div>		
<div><div>*</div><div>Update interval</div><div>60s</div></div>		

12 Descoberta usando dados do Prometheus

Visão geral

Os dados fornecidos no formato de linha do Prometheus podem ser usados para descoberta.

Consulte [verificações do Prometheus](#) para detalha como a consulta de dados do Prometheus é implementada no Zabbix.

Configuração

A regra de descoberta de baixo nível deve ser criada como um **dependente item** para o mestre HTTP item que coleta dados do Prometheus.

Prometheus para JSON

Na regra de descoberta, vá para a guia Pré-processamento e selecione a opção de pré-processamento Prometheus to JSON. Os dados no formato JSON são necessários para descoberta e a opção de pré-processamento Prometheus to JSON retornará exatamente isso, com os seguintes atributos:

- nome da métrica
- valor métrico
- ajuda (se presente)
- tipo (se houver)

- etiquetas (se houver)
- linha crua

Por exemplo, consultando `wmi_logical_disk_free_bytes`:

Discovery rule
Preprocessing 1
LLD macros
Filters
Overrides

Preprocessing steps	Name	Parameters
1:	Prometheus to JSON	<code>wmi_logical_disk_free_bytes{volume=~".*"}</code>

Add

dessas linhas do Prometheus:

```
# HELP wmi_logical_disk_free_bytes Espaço livre em bytes (LogicalDisk.PercentFreeSpace)
# TYPE medidor wmi_logical_disk_free_bytes
wmi_logical_disk_free_bytes{volume="C:"} 3.5180249088e+11
wmi_logical_disk_free_bytes{volume="D:"} 2.627731456e+09
wmi_logical_disk_free_bytes{volume="HarddiskVolume4"} 4.59276288e+08
```

retornará:

```
[
  {
    "name": "wmi_logical_disk_free_bytes",
    "help": "Espaço livre em bytes (LogicalDisk.PercentFreeSpace)",
    "tipo": "medidor",
    "etiquetas": {
      "volume": "C:"
    },
    "valor": "3.5180249088e+11",
    "line_raw": "wmi_logical_disk_free_bytes{volume=\"C:\"} 3.5180249088e+11"
  },
  {
    "name": "wmi_logical_disk_free_bytes",
    "help": "Espaço livre em bytes (LogicalDisk.PercentFreeSpace)",
    "tipo": "medidor",
    "etiquetas": {
      "volume": "D:"
    },
    "valor": "2.627731456e+09",
    "line_raw": "wmi_logical_disk_free_bytes{volume=\"D:\"} 2.627731456e+09"
  },
  {
    "name": "wmi_logical_disk_free_bytes",
    "help": "Espaço livre em bytes (LogicalDisk.PercentFreeSpace)",
    "tipo": "medidor",
    "etiquetas": {
      "volume": "HarddiskVolume4"
    },
    "valor": "4.59276288e+08",
    "line_raw": "wmi_logical_disk_free_bytes{volume=\"HarddiskVolume4\"} 4.59276288e+08"
  }
]
```

Mapeamento de macros LLD

Em seguida, você deve ir para a guia de macros LLD e fazer o seguinte mapeamentos:

```
{#VOLUME}=${.labels['volume']}
{#METRIC}=${.name}
{#HELP}=${.help}
```

Protótipo de item

Você pode querer criar um protótipo de item como este:

Item prototype

Tags

Preprocessing

*

 Name

Free bytes on {#VOLUME}

Type

Dependent item

*

 Key

wmi[{#METRIC},{#VOLUME}]

Select

*

 Master item

My host: HTTP master item

Select

Type of information

Numeric (float)

Units

B

*

 History storage period

Do not keep history

Storage period

90d

*

 Trend storage period

Do not keep trends

Storage period

365d

Value mapping

type here to search

Select

Description

{#HELP}

Create enabled

☒

Discover

☒

Add

Test

Cancel

com opções de pré-processamento:

Item prototype

Tags

Preprocessing 1

Preprocessing steps

Name

Parameters

1:

Prometheus pattern

{#METRIC}{volume="{#VOLUME}"}

Add

13 Descoberta de dispositivos de bloco

De maneira semelhante ao [arquivo sistemas](#) são descobertos, é possível também descobrir dispositivos de bloco e seus tipo.

Chave do item

A chave do item a ser usada na [descoberta regra](#) é

`vfs.dev.discovery`

Este item é suportado apenas em plataformas Linux, desde o agente Zabbix 4.4.

Você pode criar regras de descoberta usando este item de descoberta e:

- filtro: **{#DEVNAME} corresponde a sd[\D]\$** - para descobrir dispositivos chamado "sd0", "sd1", "sd2", ...
- filtro: **{#DEVTYPE} corresponde a disk E {#DEVNAME} não corresponde ^loop.*** - para descobrir dispositivos de tipo de disco cujo nome não comece com "loop"

Macros compatíveis

Esta chave de descoberta retorna duas macros - {#DEVNAME} e {#DEVTYPE} identificando o nome e o tipo do dispositivo de bloco respectivamente, por exemplo:

```
[
  {
    "{#DEVNAME}": "loop1",
    "{#DEVTYPE}": "disco"
  },
  {
    "{#DEVNAME}": "dm-0",
    "{#DEVTYPE}": "disco"
  },
  {
    "{#DEVNAME}": "sda",
    "{#DEVTYPE}": "disco"
  },
  {
    "{#DEVNAME}": "sda1",
    "{#DEVTYPE}": "partição"
  }
]
```

A descoberta de dispositivo de bloco permite usar `vfs.dev.read[]` e itens `vfs.dev.write[]` para criar protótipos de itens usando o {#DEVNAME} macro, por exemplo:

- "vfs.dev.read[{#DEVNAME},sps]"
- "vfs.dev.write[{#DEVNAME},sps]"

{#DEVTYPE} destina-se à filtragem de dispositivos.

14 Descoberta de interfaces de host no Zabbix

Visão geral

É possível **descobrir** todos interfaces configuradas no frontend do Zabbix para um host.

Chave do item

O item a ser usado na **descoberta regra** é o

`zabbix[host, descoberta, interfaces]`

artigo interno. Este item é suportado desde o servidor Zabbix 3.4.

Este item retorna um JSON com a descrição das interfaces, incluindo:

- Endereço IP/nome do host DNS (dependendo da configuração do host "Connect to")
- Número da porta
- Tipo de interface (agente Zabbix, SNMP, JMX, IPMI)
- Se é a interface padrão ou não
- Se o recurso de solicitação em massa estiver ativado - apenas para interfaces SNMP.

Por exemplo:

```
[{"{#IF.CONN}": "192.168.3.1", "{#IF.IP}": "192.168.3.1", "{#IF.DNS}": "", "{#IF.PORT }": "10050", "{#IF.TYPE}": "A"
```

Com várias interfaces, seus registros em JSON são ordenados por:

- Tipo de interface,
- Padrão - a interface padrão é colocada antes do não padrão interfaces,
- ID da interface (em ordem crescente).

Macros compatíveis

As seguintes macros têm suporte para uso na regra de descoberta **filtro** e protótipos de itens, gatilhos e gráficos:

Macro	Descrição
{#IF.CONN}	Endereço IP da interface ou nome do host DNS.
{#IF.IP}	Endereço IP da interface.
{#IF.DNS}	Nome do host DNS da interface.
{#IF.PORT}	Número da porta da interface.
{#IF.TYPE}	Tipo de interface ("AGENT", "SNMP", "JMX" ou "IPMI").
{#IF.DEFAULT}	Status padrão da interface: 0 - não interface padrão 1 - interface padrão
{#IF.SNMP.BULK}	Status de processamento em massa SNMP para a interface: 0 - desabilitado 1 - habilitado Esta macro é retornada somente se o tipo de interface for "SNMP".

6 Regras de LLD personalizadas

Visão geral

Também é possível criar uma regra de LLD completamente personalizada, descobrindo qualquer tipo de entidades - por exemplo, bancos de dados em um servidor de banco de dados.

Para fazer isso, um item personalizado deve ser criado para que retorne JSON, especificando objetos encontrados, e opcionalmente, algumas de suas propriedades. A quantidade de macros por entidade não é limitada - enquanto as regras de criação retornam uma ou duas macros (por exemplo, duas para descobertas de sistema de arquivos) é possível retornar mais.

Exemplo

O formato JSON necessário é melhor ilustrado com um exemplo. Digamos que estamos executando um agente Zabbix 1.8 antigo (um que não suporta "vfs.fs.discovery"), mas ainda precisamos descobrir os sistemas de arquivo. Aqui está um simples script Perl para Linux que descobre sistemas de arquivos montados e gera um JSON, que inclui tanto o nome quanto o tipo do sistema de arquivos. Uma forma de usar seria com um UserParameter com key "vfs.fs.discovery_perl":

```
####!/usr/bin/perl

$first = 1;

print "[\n";

for (`cat /proc/mounts`)
{
    ($fsname, $fstype) = m/\S+ (\S+) (\S+)/;

    print "\t,\n" if not $first;
    $first = 0;

    print "\t{\n";
    print "\t\t\t\"{#FSNAME}\" : \"$fsname\", \n";
    print "\t\t\t\"{#FSTYPE}\" : \"$fstype\" \n";
    print "\t}\n";
}

print "]\n";
```

Attention:

Os símbolos permitidos para nomes macros são **0-9**, **A-Z**, **_**, **.** Letras minúsculas não são permitidas nos nomes.

Um exemplo de seu output (reformatado para um melhor entendimento) é mostrado abaixo. O JSON para verificações personalizadas de descoberta tem que seguir o mesmo formato abaixo.

```
[
  { "#{FSNAME}":"/", "FSTYPE":"rootfs" },
  { "#{FSNAME}":"/sys", "FSTYPE":"sysfs" },
  { "#{FSNAME}":"/proc", "FSTYPE":"proc" },
  { "#{FSNAME}":"/dev", "FSTYPE":"devtmpfs" },
  { "#{FSNAME}":"/dev/pts", "FSTYPE":"devpts" },
  { "#{FSNAME}":"/lib/init/rw", "FSTYPE":"tmpfs" },
  { "#{FSNAME}":"/dev/shm", "FSTYPE":"tmpfs" },
  { "#{FSNAME}":"/home", "FSTYPE":"ext3" },
  { "#{FSNAME}":"/tmp", "FSTYPE":"ext3" },
  { "#{FSNAME}":"/usr", "FSTYPE":"ext3" },
  { "#{FSNAME}":"/var", "FSTYPE":"ext3" },
  { "#{FSNAME}":"/sys/fs/fuse/connections", "FSTYPE":"fusectl" }
]
```

No exemplo anterior, é necessário que as keys correspondam aos nomes das macros de LLD usadas em prototypes. A alternativa é extrair os valores das macros de LLD usando JSONPath `{#FSNAME} → $.fsname` e `{#FSTYPE} → $.fstype`, tornando assim possível um script como este:

```
####!/usr/bin/perl

$first = 1;

print "[\n";

for (`cat /proc/mounts`)
{
    ($fsname, $fstype) = m/\S+ (\S+) (\S+)/;

    print "\t,\n" if not $first;
    $first = 0;

    print "\t{\n";
    print "\t\t\"fsname\": \"$fsname\", \n";
    print "\t\t\"fstype\": \"$fstype\", \n";
    print "\t}\n";
}

print "]\n";
```

Um exemplo de seus outputs (reformatados para um melhor entendimento) é mostrado a seguir. O JSON para verificações personalizadas deve seguir o mesmo formato abaixo.

```
[
  { "fsname":"/", "fstype":"rootfs" },
  { "fsname":"/sys", "fstype":"sysfs" },
  { "fsname":"/proc", "fstype":"proc" },
  { "fsname":"/dev", "fstype":"devtmpfs" },
  { "fsname":"/dev/pts", "fstype":"devpts" },
  { "fsname":"/lib/init/rw", "fstype":"tmpfs" },
  { "fsname":"/dev/shm", "fstype":"tmpfs" },
  { "fsname":"/home", "fstype":"ext3" },
  { "fsname":"/tmp", "fstype":"ext3" },
  { "fsname":"/usr", "fstype":"ext3" },
  { "fsname":"/var", "fstype":"ext3" },
  { "fsname":"/sys/fs/fuse/connections", "fstype":"fusectl" }
]
```

Em seguida, no campo "Filter" da regra de descoberta, poderíamos especificar `"#{FSTYPE}"` como uma macro e `"rootfs|ext3"` como uma expressão regular.

Note:

Você não precisa utilizar os nomes macros FSNAME/FSTYPE com regras de LLD personalizadas, você tem liberdade para utilizar os nomes que desejar. Caso JSONPath seja utilizado, a linha LLD será um elemento array que pode ser um objeto, mas também pode ser uma outra array ou um valor.

Observe que, ao usar um parâmetro de usuário, o valor retornado será limitado a 16MB. Para mais detalhes, ver [data limits for LLD return values](#).

16 Monitoração distribuída

Visão geral O Zabbix fornece um método eficiente e confiável para monitorar infraestrutura de TI distribuída: os [proxies](#).

Os proxies podem ser utilizados para coletar dados locais em nome de um Zabbix Server centralizado, transmitindo periodicamente os dados para a central.

Recursos de Proxy

Ao optar por usar/não um proxy, várias considerações deverão ser levadas em conta.

	Proxy
Leve	Sim
Interface gráfica	No
Serviços independentes	Sim
Fácil manutenção	Sim
Criação automática de banco de dados ¹	Sim
Administração local	No
Pronto para hardware incorporado	Sim
Conexões TCP unidirecionais	Sim
Configuração centralizada	Sim
Gera notificações	Não

Note:

[1] O recurso de criação automática de banco de dados só funciona com SQLite. Outros bancos de dados requerem uma [configuração manual](#).

Proxies

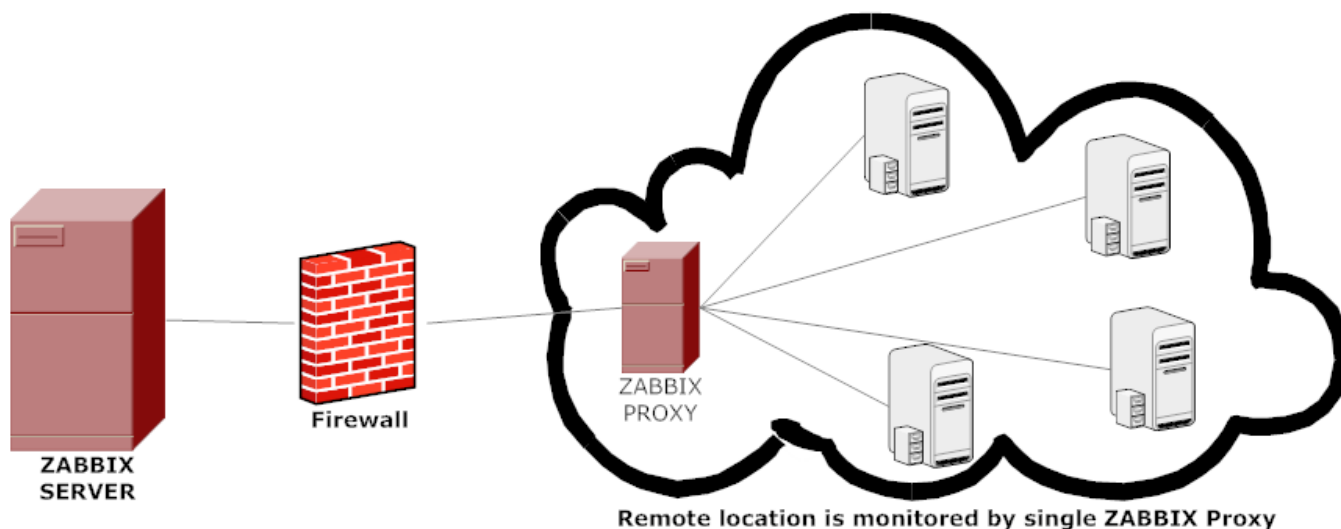
Visão geral

Um Proxy Zabbix pode coletar dados de performance e disponibilidade em nome do Zabbix Server. Desta forma o proxy transfere para si parte da carga de processamento de coletar os dados que normalmente seria atribuída ao Zabbix Server.

Além disso, usar um proxy é a forma mais fácil de implementar uma monitoração com configuração centralizada mas coleta distribuída, onde os agentes e proxies se reportam a um Zabbix Server e os dados são armazenados de forma centralizada.

Um Zabbix proxy pode ser usado para:

- Monitorar localizações remotas
- Monitorar localizações com conexão instável
- Reduzir a carga de processamento no Zabbix Server quando este tem que monitorar milhares de dispositivos
- Simplificar a manutenção da monitoração distribuída



O proxy requer apenas uma conexão com o Zabbix Server. Desta forma fica mais simples de se configurar as regras de firewall.

Attention:

O Zabbix proxy precisa utilizar um banco de dados em separado. Se ele for configurado para usar o mesmo BD do Zabbix Server irá corromper a configuração.

Todos os dados coletados pelo proxy são armazenados de forma local temporariamente antes de transmiti-los para o Zabbix Server. Desta forma nenhum dado será perdido se ocorrerem problemas temporários de comunicação entre as partes. Os parâmetros ProxyLocalBuffer e ProxyOfflineBuffer no [arquivo de configuração do proxy](#) controlam a quantidade de dados que pode ser guardada localmente.

O Zabbix proxy é um coletor de dados. Ele não calcula triggers, processa eventos ou envia alertas. Para uma visão geral do que um proxy é capaz de fazer, revise a tabela a seguir:

Funcionalidade	Suportado pelo proxy
Itens	
Zabbix agent checks (passivo)	Sim
Zabbix agent checks (ativo)	Sim ¹
Simple checks	Sim
Trapper items	Sim
SNMP checks	Sim
SNMP traps	Sim
IPMI checks	Sim
JMX checks	Sim
Monitoração de logs	Sim
Verificações internas	Sim
SSH checks	Sim
Telnet checks	Sim
External checks	Sim
Monitoração web	Sim
Descoberta de rede	Sim
Autobusca	Sim
Calcular triggers	Não
Processar eventos	Não
Enviar alertas	Não
Comandos remotos	Não

Note:

[1] Para se certificar que um agente irá se comunicar com o proxy (e não com o servidor) para a monitoração ativa, o proxy deverá estar listado no parâmetro **ServerActive** do arquivo de configuração do agente.

Protection from overloading

If Zabbix server was down for some time, and proxies have collected a lot of data, and then the server starts, it may get overloaded (history cache usage stays at 95-100% for some time). This overload could result in a performance hit, where checks are processed

slower than they should. Protection from this scenario was implemented to avoid problems that arise due to overloading history cache.

When Zabbix server history cache is full the history cache write access is being throttled, stalling server data gathering processes. The most common history cache overload case is after server downtime when proxies are uploading gathered data. To avoid this proxy throttling was added (currently it cannot be disabled).

Zabbix sever will stop accepting data from proxies when history cache usage reaches 80%. Instead those proxies will be put on a throttling list. This will continue until the cache usage falls down to 60%. Now server will start accepting data from proxies one by one, defined by the throttling list. This means the first proxy that attempted to upload data during the throttling period will be served first and until it's done the server will not accept data from other proxies.

This throttling mode will continue until either cache usage hits 80% again or falls down to 20% or the throttling list is empty. In the first case the server will stop accepting proxy data again. In the other two cases the server will start working normally, accepting data from all proxies.

You may use the `zabbix[wcache,history,pused]` internal item to correlate this behavior of Zabbix server with a metric.

Configuração

Após **Instalado** e **Configurado** um proxy, está na hora de configurá-lo no frontend do Zabbix.

Adicionando proxies

Para configurar um proxy:

- Acesse Administração → Proxies
- Clique no botão Criar proxy

Proxy

Encryption

* Proxy name

Remote proxy

Proxy mode

Active

Passive

Proxy address

127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com

Description

Add

Cancel

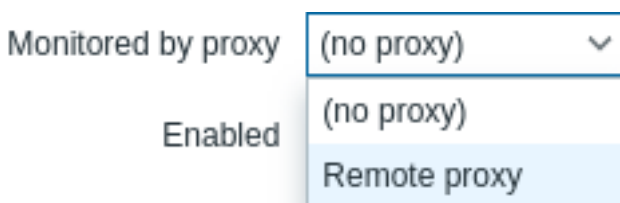
Parâmetro	Descrição
Nome do proxy	Informe o nome do proxy. Precisa ser o mesmo que o retornado pelo parâmetro Hostname pelo arquivo de configuração do proxy.
Modo do Proxy	Selecione o modo do proxy. Ativo - o proxy irá se conectar ao Zabbix Server e solicitar os dados de configuração Passivo - O Zabbix server que se conecta ao proxy Nota observe que dados sensíveis de configuração não podem se tornar disponíveis a partes que tenham acesso à porta de 'trapper' do Zabbix Server. Isso por que ninguém pode fingir ser um conjunto de dados de proxy e um pedido de comunicação ativa, isso não ocorre.
Hosts	Adiciona hosts para serem monitorados pelo proxy.
Descrição	Descrição do proxy.

A aba **Criptografia** permite que as comunicações com o proxy sejam mais seguras.

Parâmetro	Descrição
Conexões para o host	Como o Zabbix Server irá se conectar a um proxy passivo: Sem criptografia (padrão), PSK (pre-shared key) ou Certificado.
Conexões do proxy	Quais tipos de conexão a partir deste proxy serão aceitas. Vários tipos de conexão podem ser selecionados ao mesmo tempo (muito útil para teses e modificação de tipo de conexão). O padrão é "Sem criptografia".
Emissor	Emissor autorizado para o certificado. O certificado será validado primariamente com a CA (autoridade certificadora). Se for válido e assinado pela CA, então o campo then the Emissor poderá ser utilizado para restringir as CAs. Este campo é opcional e objetiva que sua instalação Zabbix suporte certificados de múltiplas autoridades certificadoras (CAs).
Assunto	Assunto permitido no certificado. O certificado será primerio validado com a CA, se válido e assinado então o campo Assunto poderá ser usado para só permitir certificados com determinado texto no assunto. Se este campo estiver vazio qualquer certificado assinado pela CA será aceito.
Identidade PSK	Texto de identidade PSK Pre-shared key.
PSK	Pre-shared key (texto hexadecimal). Tamanho máximo: 512 hex-digits (256-byte PSK) se o Zabbix utilizar as bibliotecas GnuTLS ou OpenSSL, 64 hex-digits (32-byte PSK) se o Zabbix utilizar a biblioteca TLS (PolarSSL). Exemplo: 1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c54731949

Configuração do host

Você pode associar a monitoração de um host para que seja feita através de um proxy simplesmente modificando a **configuração do host**, usando o campo Monitorado pelo proxy.



O processo de **atualização em massa** de hosts é outra forma de especificar quais hosts serão monitorados por um proxy.

17. Criptografia

Visão geral Zabbix suporta comunicações criptografadas entre componentes Zabbix usando Protocolo Transport Layer Security (TLS) v.1.2 e 1.3 (dependendo do biblioteca de criptografia). Criptografia baseada em certificado e baseada em chave pré-compartilhada é suportado.

A criptografia pode ser configurada para conexões:

- Entre servidor Zabbix, proxy Zabbix, agente Zabbix, zabbix_sender e utilitários zabbix_get
- Para o banco de dados Zabbix **do frontend Zabbix e servidor/proxy**

A criptografia é opcional e configurável para componentes individuais:

- Alguns proxies e agentes podem ser configurados para usar certificados criptografia com o servidor, enquanto outros podem usar criptografia baseada em chave, e ainda outros continuam com não criptografado comunicações (como antes)
- Servidor (proxy) pode usar diferentes configurações de criptografia para diferentes anfitriões

Os programas daemon Zabbix usam uma porta de escuta para conexões de entrada não criptografadas. Adicionar uma criptografia não requer abrindo novas portas em firewalls.

Limitações

- As chaves privadas são armazenadas em texto simples em arquivos legíveis pelo Zabbix componentes durante a inicialização

- Chaves pré-compartilhadas são inseridas no frontend do Zabbix e armazenadas no Zabbix banco de dados em texto simples
- A criptografia integrada não protege as comunicações:
 - Entre o servidor web rodando o frontend Zabbix e a web do usuário navegador
 - Entre o frontend Zabbix e o servidor Zabbix
- Atualmente, cada conexão criptografada é aberta com um handshake TLS completo, nenhum cache de sessão e tickets são implementados
- A adição de criptografia aumenta o tempo para verificações e ações de itens, dependendo da latência da rede:
 - Por exemplo, se o atraso do pacote for de 100ms, abrir um TCP conexão e envio de solicitação não criptografada leva cerca de 200ms. Com criptografia são adicionados cerca de 1000 ms para estabelecer o TLS conexão;
 - Os tempos limite podem precisar ser aumentados, caso contrário, alguns itens e ações que executam scripts remotos em agentes podem funcionar com conexões não criptografadas, mas falham com o tempo limite com criptografado.
- A criptografia não é suportada por **network descoberta**. Verificações do agente Zabbix realizada pela descoberta de rede não será criptografada e se o Zabbix agente está configurado para rejeitar conexões não criptografadas, tais verificações não terá sucesso.

Compilando Zabbix com suporte a criptografia Para suportar a criptografia, o Zabbix deve ser compilado e vinculado a um dos bibliotecas de criptografia suportadas:

- GnuTLS - da versão 3.1.18
- OpenSSL - versões 1.0.1, 1.0.2, 1.1.0, 1.1.1
- LibreSSL - testado com as versões 2.7.4, 2.8.2:
 - LibreSSL 2.6.x não é suportado
 - O LibreSSL é suportado como um substituto compatível do OpenSSL; as novas funções de API específicas do LibreSSL `tls_*` não são usadas. Componentes Zabbix compilados com LibreSSL não poderão usar PSK, somente certificados podem ser usados.

A biblioteca é selecionada especificando a respectiva opção para script "configurar":

- `--with-gnutls [=DIR]`
- `--with-openssl [=DIR]` (também usado para LibreSSL)

Por exemplo, para configurar as origens para servidor e agente com OpenSSL você pode usar algo como:

```
./configure --enable-server --enable-agent --with-mysql --enable-ipv6 --with-net-snmp --with-libcurl --with-openssl
```

Diferentes componentes Zabbix podem ser compilados com diferentes criptomoedas bibliotecas (por exemplo, um servidor com OpenSSL, um agente com GnuTLS).

::: não importante Se você planeja usar chaves pré-compartilhadas (PSK), considere usar bibliotecas GnuTLS ou OpenSSL 1.1.0 (ou mais recente) em Componentes Zabbix usando PSKs. Bibliotecas GnuTLS e OpenSSL 1.1.0 suporta conjuntos de cifras PSK com **Perfect Forward Sigilo**. Versões mais antigas da biblioteca OpenSSL (1.0.1, 1.0.2c) também suportam PSKs, mas os conjuntos de cifras PSK disponíveis não fornecem Perfect Forward Segredo. :::

Gerenciamento de criptografia de conexão Conexões no Zabbix podem usar:

- sem criptografia (padrão)
- **baseado em certificado RSA criptografia**
- **criptografia baseada em PSK**

Existem dois parâmetros importantes usados para especificar a criptografia entre Componentes do Zabbix:

- TLSConnect - especifica qual criptografia usar para saída conexões (não criptografadas, PSK ou certificado)
- TLSAccept - especifica quais tipos de conexões são permitidas conexões de entrada (não criptografadas, PSK ou certificado). Um ou mais valores podem ser especificados.

TLSConnect é usado nos arquivos de configuração do proxy Zabbix (no modo ativo, especifica apenas conexões com o servidor) e agente Zabbix (para verificações ativas). No frontend do Zabbix, o equivalente do TLSConnect é o Conexões ao host campo em Configuração → Hosts → <alguns host> → guia Criptografia e o campo Conexões ao proxy em Administração → Proxies → <algum proxy> → guia Criptografia. Se o tipo de criptografia configurado para falha de conexão, nenhuma outra criptografia tipos serão tentados.

TLSAccept é usado nos arquivos de configuração do proxy Zabbix (no modo passivo, especifica apenas conexões do servidor) e agente Zabbix (para verificações passivas). No frontend do Zabbix, o equivalente do TLSAccept é o Conexões do host campo em Configuração → Hosts → <alguns host> → aba Criptografia e o campo Conexões do proxy em Administração → Proxies → <algum proxy> → guia Criptografia.

Normalmente você configura apenas um tipo de criptografia para criptografias. Mas você pode querer mudar o tipo de criptografia, por exemplo a partir de descriptografado para baseado em certificado com tempo de inatividade e reversão mínimos possibilidade. Para alcançar isto:

- Defina `TLSAccept=unencrypted`, cert no arquivo de configuração do agente e reinicie o agente Zabbix
- Teste a conexão com o `zabbix_get` para o agente usando o certificado. Se funciona, você pode reconfigurar a criptografia para esse agente no Zabbix frontend na Configuração → Hosts → <algum host> → Aba Criptografia definindo Conexões ao host para "Certificado".
- Quando o cache de configuração do servidor é atualizado (e proxy configuração é atualizada se o host for monitorado por proxy) então as conexões com esse agente serão criptografadas
- Se tudo funcionar como esperado, você pode definir `TLSAccept=cert` no arquivo de configuração do agente e reinicie o agente Zabbix. Agora o agente aceitará apenas conexões baseadas em certificados criptografados. Conexões não criptografadas e baseadas em PSK serão rejeitadas.

De maneira semelhante, funciona no servidor e no proxy. Se no Zabbix frontend em configuração do host Conexões do host é definido como "Certificado" e somente conexões criptografadas baseadas em certificado serão aceitas do agent (verificações ativas) e `zabbix_sender` (itens do trapper).

Muito provavelmente você irá configurar conexões de entrada e saída para usar o mesmo tipo de criptografia ou nenhuma criptografia. Mas tecnicamente é possível configurá-lo assimetricamente, por ex. baseado em certificado criptografia para conexões de entrada e baseadas em PSK para conexões de saída.

A configuração de criptografia para cada host é exibida no Zabbix frontend, em Configuração → Hosts na coluna Criptografia do agente. Por exemplo:

Exemplo	Conexões ao host	Conexões permitidas do host	Conexões rejeitadas do host
	NONE	Unencrypted	Unencrypted Criptografado, certificado e criptografado com base em PSK
CERT	NONE PSK CERT	Criptografado, baseado em certificado	Criptado, baseado em certificado Não criptografado e criptografado com base em PSK
	PSK NONE PSK CERT	Criptografado, baseado em PSK	Criptado, baseado em PSK Não criptografado e criptografado com base em certificado
	PSK NONE PSK CERT	Criptografado, baseado em PSK	Criptado, baseado em PSK Criptografia não criptografada e baseada em PSK Criptografia baseada em certificado
	CERT NONE PSK CERT	Criptografado, baseado em certificado	Criptado, baseado em certificado Não criptografado, PSK ou criptografado baseado em certificado

::: não importante As conexões não são criptografadas por padrão. Criptografia deve ser configurado para cada host e proxy individualmente. :::

zabbix_get e zabbix_sender com criptografia Veja [zabbix_get](#) e [zabbix_sender](#) manpages para usá-los com criptografia.

Ciphersuites Ciphersuites por padrão são configurados internamente durante a inicialização do Zabbix e, antes do Zabbix 4.0.19, 4.4.7, não são configuráveis pelo usuário.

Desde o Zabbix 4.0.19, 4.4.7 também os ciphersuites configurados pelo usuário são suportado para GnuTLS e OpenSSL. Os usuários podem **configure** ciphersuites de acordo com suas políticas de segurança. O uso deste recurso é opcional (integrados ciphersuites padrão ainda funcionam).

Para bibliotecas de criptografia compiladas com configurações padrão Zabbix built-in regras normalmente resultam nos seguintes conjuntos de cifras (em ordem de prioridade mais alta para mais baixa):

Biblioteca	Certificate ciphersuites	PSK ciphersuites
GnuTLS 3.1.18	TLS_ECDHE_RSA_AES_128_GCM_SHA256	TLS_ECDHE_PSK_AES_128_CBC_SHA256
	TLS_ECDHE_RSA_AES_128_CBC_SHA256	TLS_ECDHE_PSK_AES_128_CBC_SHA1
	TLS_ECDHE_RSA_AES_128_CBC_SHA1	TLS_PSK_AES_128_GCM_SHA256
	TLS_RSA_AES_128_GCM_SHA256	TLS_PSK_AES_128_CBC_SHA256
	TLS_RSA_AES_128_CBC_SHA256	TLS_PSK_AES_128_CBC_SHA1
	TLS_RSA_AES_128_CBC_SHA1	
OpenSSL 1.0.2c	ECDHE-RSA-AES128-GCM-SHA256	PSK-AES128-CBC-SHA
	ECDHE-RSA-AES128-SHA256	
	ECDHE-RSA-AES128-SHA	
	AES128-GCM-SHA256	
	AES128-SHA256	
	AES128-SHA	

Biblioteca	Certificate ciphersuites	PSK ciphersuites
OpenSSL 1.1.0	ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-CCM8 AES128-CCM AES128-SHA256 AES128-SHA	ECDHE-PSK-AES128-CBC-SHA256 ECDHE-PSK-AES128-CBC-SHA PSK-AES128-GCM-SHA256 PSK-AES128-CCM8 PSK-AES128-CCM PSK-AES128-CBC-SHA256 PSK-AES128-CBC-SHA
OpenSSL 1.1.1d	TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-CCM8 AES128-CCM AES128-SHA256 AES128-SHA	TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256 ECDHE-PSK-AES128-CBC-SHA256 ECDHE-PSK-AES128-CBC-SHA PSK-AES128-GCM-SHA256 PSK-AES128-CCM8 PSK-AES128-CCM PSK-AES128-CBC-SHA256 PSK-AES128-CBC-SHA

Ciphersuites configurados pelo usuário Os critérios de seleção do conjunto de cifras integrados podem ser substituídos por ciphersuites configurados pelo usuário.

::: não importante Ciphersuites configurados pelo usuário é um recurso destinado para usuários avançados que entendem conjuntos de criptografia TLS, sua segurança e consequências de erros e que se sentem à vontade com o TLS solução de problemas.
:::

Os critérios de seleção do ciphersuite integrados podem ser substituídos usando o seguintes parâmetros:

Substituir escopo	Parâmetro	Valor	Descrição
Seleção de Ciphersuite para certificados	TLSCipherCert13	Válido OpenSSL 1.1.1 cipher strings para protocolo TLS 1.3 (seus valores são passados para a função OpenSSL SSL_CTX_set_ciphersuites()).	Critérios de seleção de conjuntos de códigos baseados em certificados para TLS 1.3 Somente OpenSSL 1.1.1 ou mais recente.
	TLSCipherCert	OpenSSL válido strings de criptografia para TLS 1.2 ou GnuTLS válido strings de prioridade . Seus valores são passados para as funções SSL_CTX_set_cipher_list() ou gnutls_priority_init(), respectivamente.	Critérios de seleção de conjuntos de códigos baseados em certificados para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Substituir escopo	Parâmetro	Valor	Descrição
Seleção de Ciphersuite para PSK	TLSCipherPSK13	Válido OpenSSL 1.1.1 cipher strings para protocolo TLS 1.3 (seus valores são passados para a função OpenSSL SSL_CTX_set_ciphersuites()).	Critérios de seleção de conjuntos de códigos baseados em PSK para TLS 1.3 Somente OpenSSL 1.1.1 ou mais recente.
	TLSCipherPSK	OpenSSL válido strings de criptografia para TLS 1.2 ou GnuTLS válido strings de prioridade . Seus valores são passados para as funções SSL_CTX_set_cipher_list() ou gnutls_priority_init(), respectivamente.	Critérios de seleção de ciphersuite baseados em PSK para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)
Lista combinada de ciphersuite para certificado e PSK	TLSCipherAll13	Valid OpenSSL 1.1.1 cipher strings para protocolo TLS 1.3 (seus valores são passados para a função OpenSSL SSL_CTX_set_ciphersuites()).	Critérios de seleção de Ciphersuite para TLS 1.3 Somente OpenSSL 1.1.1 ou mais recente.
	TLSCipherAll	OpenSSL válido strings de criptografia para TLS 1.2 ou GnuTLS válido strings de prioridade . Seus valores são passados para as funções SSL_CTX_set_cipher_list() ou gnutls_priority_init(), respectivamente.	Critérios de seleção do Ciphersuite para TLS 1.2/1.3 (GnuTLS), TLS 1.2 (OpenSSL)

Para substituir a seleção do ciphersuite em [zabbix_get](#) e [zabbix_sender](#) utilitários - use o parâmetros de linha de comando:

- `--tls-cipher13`
- `--tls-cipher`

Os novos parâmetros são opcionais. Se um parâmetro não for especificado, o valor padrão interno é usado. Se um parâmetro é definido, ele não pode ser vazio.

Se a configuração de um valor TLSCipher* na biblioteca de criptografia falhar, o servidor, proxy ou agente não será iniciado e um erro será registrado.

É importante entender quando cada parâmetro é aplicável.

Conexões de saída

O caso mais simples são as conexões de saída:

- Para conexões de saída com certificado - use TLS cipherCert13 ou TLS cipherCert
- Para conexões de saída com PSK - use TLS cipherPSK13 e TLS cipherPSK
- No caso dos utilitários zabbix_get e zabbix_sender a linha de comando parâmetros --tls-cipher13 e --tls-cipher podem ser usados (a criptografia é especificada sem ambiguidade com um --tls-connect parâmetro)

Conexões de saída

O caso mais simples são as conexões de saída:

- Para conexões de saída com certificado - use TLS cipherCert13 ou TLS cipherCert
- Para conexões de saída com PSK - use TLS cipherPSK13 e TLS cipherPSK
- No caso dos utilitários zabbix_get e zabbix_sender a linha de comando parâmetros --tls-cipher13 e --tls-cipher podem ser usados (a criptografia é especificada sem ambiguidade com um --tls-connect parâmetro)

Testando strings de cifras e permitindo apenas conjuntos de cifras PFS

Para ver quais ciphersuites foram selecionados, você precisa definir 'DebugLevel=4' no arquivo de configuração, ou use a opção -vv para zabbix_sender.

Algumas experiências com os parâmetros TLS cipher* podem ser necessárias antes de obter os conjuntos de cifras desejados. É inconveniente reiniciar Servidor, proxy ou agente Zabbix várias vezes apenas para ajustar TLS cipher* parâmetros. Opções mais convenientes estão usando zabbix_sender ou o comando openssl. Vamos mostrar os dois.

1. Usando zabbix_sender.

Vamos fazer um arquivo de configuração de teste, por exemplo /home/zabbix/test.conf, com a sintaxe de um arquivo zabbix_agentd.conf:

```
Nome do host=não existente
ServerActive=inexistente

TLSConnect=cert
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/agent.crt
TLSKeyFile=/home/zabbix/agent.key
TLSPSKIdentity=inexistente
TLSPSKFile=/home/zabbix/agent.psk
```

Você precisa de certificados válidos de CA e agente e PSK para este exemplo. Ajuste os caminhos e nomes de certificados e arquivos PSK para seu ambiente.

Se você não estiver usando certificados, mas apenas PSK, você pode simplificar arquivo de teste:

```
Nome do host=não existente
ServerActive=inexistente

TLSConnect=psk
TLSPSKIdentity=inexistente
TLSPSKFile=/home/zabbix/agentd.psk
```

Os ciphersuites selecionados podem ser vistos executando zabbix_sender (exemplo compilado com OpenSSL 1.1.1.d):

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k item_não-existente -o 1 2>&1 | grep ciphersuites
zabbix_sender [41271]: depuração: zbx_tls_init_child () ciphersuites de certificados: TLS_AES_256_GCM_SHA384
zabbix_sender [41271]: depuração: zbx_tls_init_child () PSK ciphersuites: TLS_CHACHA20_POLY1305_SHA256
zabbix_sender [41271]: depuração: zbx_tls_init_child () certificado e PSK ciphersuites: TLS_AES_256_GCM_SHA384
```

Aqui você vê os conjuntos de cifras selecionados por padrão. Esses valores padrão são escolhidos para garantir a interoperabilidade com os agentes Zabbix rodando em sistemas com versões mais antigas do OpenSSL (de 1.0.1).

Com sistemas mais novos, você pode optar por aumentar a segurança permitindo apenas alguns conjuntos de cifras, e apenas ciphersuites com PFS (Perfect Forward Segredo). Vamos tentar permitir apenas ciphersuites com PFS usando Parâmetros `TLSCipher*`.

::: não importante O resultado não será interoperável com sistemas usando OpenSSL 1.0.1 e 1.0.2, se PSK for usado. Baseado em certificado criptografia deve funcionar. :::

Adicione duas linhas ao arquivo de configuração `test.conf`:

```
TLSCipherCert=EECDH+aRSA+AES128
TLS CipherPSK=kECDHEPSK+AES128
```

e teste novamente:

```
$ zabbix_sender -vv -c /home/zabbix/test.conf -k item_não-existente -o 1 2>&1 | grep ciphersuites
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() certificados ciphersuites: TLS_AES_256_GCM_SHA384 TLS_AES_128_GCM_SHA256
zabbix_sender [42892]: DEBUG: zbx_tls_init_child() Ciphersuites PSK: TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256
zabbix_sender [42892]: depuração: zbx_tls_init_child () certificado e PSK ciphersuites: TLS_AES_256_GCM_SHA384
```

As listas "certificate ciphersuites" e "PSK ciphersuites" foram alteradas - eles são mais curtos do que antes, contendo apenas conjuntos de cifras TLS 1.3 e TLS 1.2 ECDHE-* ciphersuites conforme esperado.

2. `TLSCipherAll` e `TLSCipherAll13` não podem ser testados com `zabbix_sender`; eles não afetam "certificados e conjuntos de cifras PSK" valor mostrado no exemplo acima. Para ajustar `TLSCipherAll` e `TLSCipherAll13` você precisa experimentar com o agente, proxy ou servidor.

Portanto, para permitir apenas conjuntos de cifras PFS, você pode precisar adicionar até três parâmetros

```
TLSCipherCert=EECDH+aRSA+AES128
TLS CipherPSK=kECDHEPSK+AES128
TLSCipherAll=EECDH+aRSA+AES128:kECDHEPSK+AES128
```

para `zabbix_agentd.conf`, `zabbix_proxy.c`

Mudando de AES128 para AES256

O Zabbix usa o AES128 como padrão interno para dados. Vamos supor que você está usando certificados e deseja mudar para AES256, no OpenSSL 1.1.1.

Isso pode ser feito adicionando os respectivos parâmetros em `zabbix_server.conf`:

```
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/server.crt
TLSKeyFile=/home/zabbix/server.key
TLSCipherCert13=TLS_AES_256_GCM_SHA384
TLSCipherCert=EECDH+aRSA+AES256:-SHA1:-SHA384
TLSCipherPSK13=TLS_CHACHA20_POLY1305_SHA256
TLSCipherPSK=kECDHEPSK+AES256:-SHA1
TLSCipherAll13=TLS_AES_256_GCM_SHA384
TLSCipherAll=EECDH+aRSA+AES256:-SHA1:-SHA384
```

::: não importante Embora apenas conjuntos de criptografia relacionados a certificados será usado, os parâmetros `TLSCipherPSK*` também são definidos para evitar seus valores padrão que incluem cifras menos seguras para interoperabilidade. Ciphersuites PSK não podem ser completamente desabilitados em servidor/proxy. :::

E em `zabbix_agentd.conf`:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/ca.crt
TLSCertFile=/home/zabbix/agent.crt
TLSKeyFile=/home/zabbix/agent.key
TLSCipherCert13=TLS_AES_256_GCM_SHA384
TLSCipherCert=EECDH+aRSA+AES256:-SHA1:-SHA384
```

1 Usando certificados

Visão geral

O Zabbix pode usar certificados RSA no formato PEM, assinados por um público ou autoridade de certificação interna (CA). A verificação do certificado é feita contra um certificado CA pré-configurado. Certificado opcional listas de revogação (CRL) podem ser usadas. Cada componente Zabbix pode ter apenas um certificado configurado.

Para obter mais informações sobre como configurar e operar CA interna, como gerar solicitações de certificado e assiná-las, como revogar certificados você pode encontrar vários tutoriais online, por exemplo, [OpenSSL PKI Tutorial v1.1](#).

Considere e teste cuidadosamente suas extensões de certificado - consulte [Limitações ao usar o certificado X.509 v3 extensões](#).

Parâmetros de configuração do certificado

Parâmetro	Obrigatório	Descrição
TLSCAFile	*	Nome do caminho completo de um arquivo contendo os certificados de CA(s) de nível superior para verificação de certificado de peer. No caso de cadeia de certificados com vários membros, eles devem ser solicitados: certificados de CA de nível inferior seguidos primeiro por certificados de CA(s) de nível superior. Certificados de várias CA(s) podem ser incluídos em um único arquivo.
TLSCRLFile		Nome completo do caminho de um arquivo contendo Listas de Revogação de Certificados. Veja as notas em Listas de revogação de certificados (CRL) .
TLSCertFile	*	Nome completo do caminho de um arquivo contendo certificado (cadeia de certificados). No caso de cadeia de certificados com vários membros, eles devem ser solicitados: primeiro servidor, proxy ou certificado de agente, seguido por CA de nível inferior certificados e depois certificados de CA(s) de nível superior.
TLSKeyFile	*	Nome do caminho completo de um arquivo contendo chave privada. Defina os direitos de acesso a este arquivo - ele deve ser lido apenas pelo usuário Zabbix.
TLSServerCertIssuer		Emissor de certificado de servidor permitido.
TLSServerCertSubject		Assunto do certificado do servidor permitido.

Configurando o certificado no servidor Zabbix

1. Para verificar os certificados de peer, o servidor Zabbix deve ter acesso para arquivar com seus certificados CA raiz autoassinados de nível superior. Por exemplo, se esperamos certificados de duas CAs raiz independentes, podemos colocar seus certificados no arquivo `/home/zabbix/zabbix_ca_file` como esta:

Certificado:

Dados:

Versão: 3 (0x2)

Número de série: 1 (0x1)

Algoritmo de assinatura: sha1WithRSAEncryption

Emissor: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Root1 CA

...

Assunto: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Root1 CA

Informações da chave pública do assunto:

Algoritmo de chave pública: rsaEncryption

Chave pública: (2048 bits)

...

Extensões X509v3:

Uso da chave X509v3: crítico

Sinal de certificado, sinal de CRL

Restrições básicas do X509v3: crítica

CA: VERDADEIRO

...

-----INICIAR CERTIFICADO-----

MIID2jCCAsKgAwIBAGIBATANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLQG

....

9wEzdN8uTrqoyU78gi12npLj08LegRKjb5hFTVm0

-----FIM DO CERTIFICADO-----

Certificado:

Dados:

Versão: 3 (0x2)

Número de série: 1 (0x1)

Algoritmo de assinatura: sha1WithRSAEncryption

Emissor: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Root2 CA

...

Assunto: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Root2 CA

Informações da chave pública do assunto:

Algoritmo de chave pública: rsaEncryption

Chave pública: (2048 bits)

....

Extensões X509v3:

Uso da chave X509v3: crítico

Sinal de certificado, sinal de CRL

Restrições básicas do X509v3: crítica

CA: VERDADEIRO

....

-----INICIAR CERTIFICADO-----

MIID3DCCAsSgAwIBAGIBATANBgkqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLQG

...

vdGNYoSfvu41GQAR5Vj5FnRJRzv5XQOZ3B6894GY1zY=

-----FIM DO CERTIFICADO-----

2. Coloque a cadeia de certificados do servidor Zabbix no arquivo, por exemplo, /home/zabbix/zabbix_server.crt:

Certificado:

Dados:

Versão: 3 (0x2)

Número de série: 1 (0x1)

Algoritmo de assinatura: sha1WithRSAEncryption

Emissor: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Assinar CA

...

Assunto: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=servidor Zabbix

Informações da chave pública do assunto:

Algoritmo de chave pública: rsaEncryption

Chave pública: (2048 bits)

...

Extensões X509v3:

Uso da chave X509v3: crítico

Assinatura digital, criptografia de chave

Restrições básicas do X509v3:

CA:FALSO

...

-----INICIAR CERTIFICADO-----

MIIECDCCAvCgAwIBAGIBATANBgkqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixk

...

h02u1GHIy46GI+xfR3LsPwFKlkTaaLaL/6aaoQ==

-----FIM DO CERTIFICADO-----

Certificado:

Dados:

Versão: 3 (0x2)

Número de série: 2 (0x2)

Algoritmo de assinatura: sha1WithRSAEncryption

Emissor: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Root1 CA

...

Assunto: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Assinar CA

```

Informações da chave pública do assunto:
  Algoritmo de chave pública: rsaEncryption
  Chave pública: (2048 bits)
...
Extensões X509v3:
  Uso da chave X509v3: crítico
  Sinal de certificado, sinal de CRL
  Restrições básicas do X509v3: crítica
  CA:TRUE, pathlen:0
...

```

```

-----INICIAR CERTIFICADO-----
MIID4TCCAsmgAwIBAgIBAjANBgkqhkiG9w0BAQUFADB+MRMwEQYKCZImiZPyLQGBo
...
dyCeWnvL7u5sd6ffo8iRny0QzbHKmQt/wUtcVivWXdMIFJMOHw==
-----FIM DO CERTIFICADO-----

```

Aqui o primeiro é o certificado do servidor Zabbix, seguido pela CA intermediária certificado.

3. Coloque a chave privada do servidor Zabbix no arquivo, por exemplo, `/home/zabbix/zabbix_server.key`:

```

-----INICIAR CHAVE PRIVADA-----
MIIEWAIBADANBgkqhkiG9w0BAQEFAASCBAQgAgEAAoIBAQC9tIXIJoVnNXDl
...
IJLkhbybBYEf47MLhffWa7XvZTY=
-----TERMINAR CHAVE PRIVADA-----

```

4. Edite os parâmetros TLS no arquivo de configuração do servidor Zabbix como este:

```

TLSCAFile=/home/zabbix/zabbix_ca_file
TLSCertFile=/home/zabbix/zabbix_server.crt
TLSKeyFile=/home/zabbix/zabbix_server.key

```

Configurando a criptografia baseada em certificado para o proxy Zabbix

1. Preparar arquivos com certificados de CA de nível superior, certificado de proxy (cadeia) e chave privada conforme descrito em [Configurando o certificado em Zabbix server](#). Edite os parâmetros `TLSCAFile`, `TLSCertFile`, `TLSKeyFile` no proxy configuração em conformidade.

2. Para o parâmetro "TLSConnect" de edição de proxy ativo:

```
TLSConnect=cert
```

Para o parâmetro "TLSAccept" de edição de proxy passivo:

```
TLSAccept=cert
```

3. Agora você tem uma configuração mínima de proxy baseada em certificado. Vocês pode preferir melhorar a segurança do proxy definindo `TLSServerCertIssuer` e parâmetros `TLSServerCertSubject` (consulte [Restringindo Emissor do certificado e Assunto](#)).

4. No arquivo de configuração de proxy final, os parâmetros TLS podem se parecer com:

```

TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file
TLSServerCertIssuer=CN=Assinando CA,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=servidor Zabbix,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_proxy.crt
TLSKeyFile=/home/zabbix/zabbix_proxy.key

```

5. Configure a criptografia para este proxy no frontend do Zabbix:

- Vá para: Administração → Proxies
- Selecione o proxy e clique na guia **Criptografia**

Nos exemplos abaixo, os campos Emissor e Assunto são preenchidos - consulte [Restringindo o emissor do certificado permitido e Assunto](#) por que e como usar esses campos.

Para proxy ativo

Proxy **Encryption**

Connections to proxy: No encryption PSK Certificate

Connections from proxy: ☐ No encryption
☐ PSK
☒ Certificate

Issuer: CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com

Subject: CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com

Update Clone Delete Cancel

Para proxy passivo

Proxy **Encryption**

Connections to proxy: No encryption PSK Certificate

Connections from proxy: ☒ No encryption
☐ PSK
☐ Certificate

Issuer: CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com

Subject: CN=Zabbix proxy,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com

Update Clone Delete Cancel

Configurando a criptografia baseada em certificado para o agente Zabbix

1. Preparar arquivos com certificados de CA de nível superior, certificado de agente (cadeia) e chave privada conforme descrito em [Configurando o certificado em Zabbix server](#). Edite os parâmetros TLSCAFile, TLSCertFile, TLSKeyFile no agente configuração em conformidade.

2. Para verificações ativas, edite o parâmetro TLSConnect:

TLSConnect=cert

Para verificações passivas, edite o parâmetro TLSAccept:

TLSAccept=cert

3. Agora você tem uma configuração mínima de agente baseada em certificado. Você pode preferir melhorar a segurança do agente configurando TLSServerCertIssuer e parâmetros TLSServerCertSubject. (consulte [Restringindo permitido Emissor do certificado e Assunto](#)).

4. No arquivo de configuração final do agente, os parâmetros TLS podem se parecer com:

```
TLSConnect=cert
TLSAccept=cert
TLSCAFile=/home/zabbix/zabbix_ca_file
TLSServerCertIssuer=CN=Assinando CA,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Proxy Zabbix,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSCertFile=/home/zabbix/zabbix_agentd.crt
TLSKeyFile=/home/zabbix/zabbix_agentd.key
```

(O exemplo assume que o host é monitorado via proxy, portanto, proxy Certificado Assunto.)

5. Configure a criptografia para este agente no frontend do Zabbix:

- Vá para: Configuração → Hosts
- Selecione o host e clique na guia **Criptografia**

No exemplo abaixo, os campos Emissor e Assunto são preenchidos - consulte [Restringindo o emissor do certificado permitido e Assunto](#) por que e como usar esses campos.

The screenshot shows the Zabbix web interface for configuring encryption. The 'Encryption' tab is selected. Under 'Connections to host', the 'Certificate' button is highlighted. Under 'Connections from host', the 'Certificate' checkbox is checked. The 'Issuer' field contains the text 'CN=Signing CA,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com'. The 'Subject' field contains the text 'CN=www01,OU=Development group,O=Zabbix SIA,DC=zabbix,DC=com'. At the bottom, there are five buttons: 'Update', 'Clone', 'Full clone', 'Delete', and 'Cancel'.

Restringindo o emissor e o assunto do certificado permitido

Quando dois componentes Zabbix (por exemplo, servidor e agente) estabelecem um TLS conexão ambos verificam os certificados um do outro. Se um colega certificado é assinado por uma CA confiável (com certificado de nível superior pré-configurado certificado em TLSCAFile), é válido, não expirou e passa alguns outras verificações, a comunicação pode prosseguir. Emissor de certificados e assunto não são verificados neste caso mais simples.

Aqui está um risco - qualquer pessoa com um certificado válido pode se passar por qualquer outra pessoa (por exemplo, um certificado de host pode ser usado para representar servidor). Isso pode ser aceitável em ambientes pequenos onde os certificados são assinados por uma CA interna dedicada e o risco de falsificação é baixo.

Se sua autoridade de certificação de nível superior for usada para emitir outros certificados que devem não ser aceito pelo Zabbix ou você deseja reduzir o risco de se passar por você pode restringir os certificados permitidos especificando seu emissor e Cadeias de assunto.

Por exemplo, você pode escrever no arquivo de configuração do proxy Zabbix:

```
TLSServerCertIssuer=CN=Assinando CA,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=servidor Zabbix,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
```

Com essas configurações, um proxy ativo não falará com o servidor Zabbix com string de Emissor ou Assunto diferente no certificado, um proxy passivo não aceitar solicitações de tal servidor.

Algumas notas sobre a correspondência de strings de emissor ou assunto:

1. As cadeias de emissor e assunto são verificadas independentemente. Ambos são opcional.
2. Caracteres UTF-8 são permitidos.
3. String não especificada significa que qualquer string é aceita.
4. As strings são comparadas "no estado em que se encontram", elas devem ser exatamente iguais às partida.
5. Curingas e regexps não são suportados na correspondência.
6. Apenas alguns requisitos de [RFC 4514 Lightweight Directory Access Protocol \(LDAP\): Representação de String de Distinto Nomes](#) são implementados:
 1. caracteres de escape '""' (U+0022), '+' U+002B, ',' U+002C, ';' U+003B, '<' U+003C, '>' U+003E, '\' U+005C em qualquer lugar corda.
 2. espaço de caracteres de escape (' ' U+0020) ou sinal de número ('#' U+0023) no início da string.
 3. espaço de caracteres de escape (' ' U+0020) no final da string.
7. A correspondência falha se um caractere nulo (U+0000) for encontrado ([RFC 4514](#) permite).
8. Requisitos do [Protocolo de Acesso ao Diretório Leve RFC 4517 \(LDAP\): Sintaxes e correspondência Regras](#) e [RFC 4518 Leve Protocolo de Acesso ao Diretório \(LDAP\): String Internacionalizada Preparação](#) não são suportados devido à quantidade de

trabalho necessária.

A ordem dos campos nas strings e formatação do Emissor e do Assunto são importante! Zabbix segue [RFC 4514](#) recomendação e usa a ordem "inversa" dos campos.

A ordem inversa pode ser ilustrada pelo exemplo:

```
TLSServerCertIssuer=CN=Assinando CA,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
TLSServerCertSubject=CN=Proxy Zabbix,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
```

Observe que começa com nível baixo (CN), prossegue para nível médio (OU, O) e termina com campos de nível superior (DC).

OpenSSL por padrão mostra os campos Emissor e Assunto do certificado em ordem "normal", dependendo das opções adicionais usadas:

```
$ openssl x509 -noout -in /home/zabbix/zabbix_proxy.crt -issuer -subject
emissor= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Grupo de desenvolvimento/CN=Assinando CA
subject= /DC=com/DC=zabbix/O=Zabbix SIA/OU=Grupo de desenvolvimento/CN=Proxy Zabbix
```

```
$ openssl x509 -noout -text -in /home/zabbix/zabbix_proxy.crt
```

Certificado:

```
...
Emissor: DC=com, DC=zabbix, O=Zabbix SIA, OU=Grupo de desenvolvimento, CN=Assinar CA
...
Assunto: DC=com, DC=zabbix, O=Zabbix SIA, OU=grupo de desenvolvimento, CN=proxy Zabbix
```

Aqui as strings de emissor e assunto começam com nível superior (DC) e terminam com campo de baixo nível (CN), espaços e separadores de campo dependem das opções usado. Nenhum desses valores corresponderá no Zabbix Issuer e Subject Campos!

::: não importante Para obter as seqüências de emissor e assunto adequadas utilizáveis no Zabbix invoque o OpenSSL com opções especiais

```
-nameopt esc_2253,esc_ctrl,utf8,dump_nostr,dump_unknown,dump_der,sep_comma_plus,dn_rev,sname:
```

:::

```
$ openssl x509 -noout -issuer -subject \
-nameopt esc_2253,esc_ctrl,utf8,dump_nostr,dump_unknown,dump_der,sep_comma_plus,dn_rev,sname \
-in /home/zabbix/zabbix_proxy.crt
emissor= CN=Assinando CA,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
assunto= CN=Proxy Zabbix,OU=Grupo de desenvolvimento,O=Zabbix SIA,DC=zabbix,DC=com
```

Agora os campos de string estão em ordem inversa, os campos são separados por vírgulas, podem ser usado em arquivos de configuração e frontend do Zabbix.

Limitações ao usar extensões de certificado X.509 v3

- **Nome alternativo do assunto (subjectAltName)** extensão.

Nomes de assunto alternativos da extensão subjectAltName (como IP endereço, endereço de e-mail) não são suportados pelo Zabbix. Apenas valor de O campo "Assunto" pode ser verificado no Zabbix (veja [Restringindo permissão Emissor do certificado e Assunto](#)).

Se o certificado usar a extensão subjectAltName, o resultado será depende de uma combinação particular de kits de ferramentas de criptografia Zabbix componentes são compilados (pode ou não funcionar, o Zabbix pode se recusam a aceitar tais certificados de pares).

- Extensão **Uso estendido de chave**.

Se usado, geralmente tanto clientAuth (cliente TLS WWW autenticação) e serverAuth (autenticação do servidor TLS WWW) são necessário.

Por exemplo, em verificações passivas, o agente Zabbix atua em um servidor TLS função, então serverAuth deve ser definido no certificado do agente. Para ativo verifica que o certificado do agente precisa que clientAuth seja definido.

GnuTLS emite um aviso em caso de violação de uso de chave, mas permite comunicação para prosseguir.

- Extensão **Restrições de nome**.

Nem todos os kits de ferramentas de criptografia o suportam. Esta extensão pode impedir Zabbix de carregar certificados de CA onde esta seção está marcada como crítico (depende do kit de ferramentas de criptografia específico).

Listas de revogação de certificados (CRL)

Se um certificado estiver comprometido, a CA pode revogá-lo incluindo na CRL. As CRLs podem ser configuradas no arquivo de configuração do servidor, proxy e agente usando o parâmetro TLSCRLFile. Por exemplo:

```
TLSCRLFile=/home/zabbix/zabbix_crl_file
```

onde zabbix_crl_file pode conter CRLs de várias CAs e se parecer com:

```

-----INÍCIO X509 CRL-----
MIIB/DCB5QIBATANBgqhkiG9w0BAQUFADCBgTETMBEGCgmSJomT8ixkArkWA2Nv
...
treZeUPjb7LSmZ3K2hpbZN7So0ZcAoHQ3GWd9npuctg=
-----END X509 CRL-----
-----INÍCIO X509 CRL-----
MIIB+TCB4gIBATANBgqhkiG9w0BAQUFADB/MRMwEQYKCZImiZPyLQGDBGYDY29t
...
CAEebS2CND3ShBedZ8YSi15906JvaDP611R51Ns=
-----END X509 CRL-----

```

O arquivo CRL é carregado apenas na inicialização do Zabbix. A atualização da CRL requer reinicialização.

::: não importante Se o componente Zabbix for compilado com OpenSSL e CRLs são usados, em seguida, cada CA de nível superior e intermediário no certificado chains devem ter uma CRL correspondente (pode estar vazia) em TLSCRLFile.

:::

2 Usando chaves pré-compartilhadas

Visão geral

Cada chave pré-compartilhada (PSK) no Zabbix é na verdade um par de:

- string de identidade PSK não secreta,
- valor da string PSK secreta.

A string de identidade PSK é uma string UTF-8 não vazia. Por exemplo, "ID PSK 001 Zabbix agentd". É um nome único pelo qual este PSK específico é referido pelos componentes Zabbix. Não coloque informações confidenciais em Cadeia de identidade PSK - é transmitida pela rede sem criptografia.

O valor PSK é uma sequência de dígitos hexadecimais difícil de adivinhar, por exemplo, "e560cb0d918d26d31b4f642181f5f570ad89a3909311".

Limites de tamanho

Existem limites de tamanho para identidade e valor PSK no Zabbix, em alguns casos, uma biblioteca de criptografia pode ter limite inferior:

[Componente]	Tamanho máximo da identidade PSK	Tamanho mínimo do valor PSK	Tamanho máximo do valor PSK
[Zabbix]	128 caracteres UTF-8	128 bits (PSK de 16 bytes, inserido como 32 dígitos hexadecimais)	2048 bits (PSK de 256 bytes, inserido como 512 dígitos hexadecimais)
[GnuTLS]	128 bytes (pode incluir caracteres UTF-8)	128 bits (PSK de 16 bytes, inserido como 32 dígitos hexadecimais)	2048 bits (PSK de 256 bytes, inserido como 512 dígitos hexadecimais)
[OpenSSL 1.0.x, 1.1.0]	127 bytes (pode incluir caracteres UTF-8)	127 bits (PSK de 16 bytes, inserido como 32 dígitos hexadecimais)	2048 bits (PSK de 256 bytes, inserido como 512 dígitos hexadecimais)
[OpenSSL 1.1.1]	127 bytes (pode incluir caracteres UTF-8)	127 bits (PSK de 16 bytes, inserido como 32 dígitos hexadecimais)	2048 bits (PSK de 256 bytes, inserido como 512 dígitos hexadecimais)
[OpenSSL 1.1.1a e posterior]	127 bytes (pode incluir caracteres UTF-8)	127 bits (PSK de 16 bytes, inserido como 32 dígitos hexadecimais)	2048 bits (PSK de 256 bytes, inserido como 512 dígitos hexadecimais)

::: não importante O frontend Zabbix permite configurar até String de identidade PSK de 128 caracteres e PSK de 2048 bits independentemente de bibliotecas de criptografia usadas.

Se alguns componentes do Zabbix suportam limites mais baixos, é do usuário responsabilidade de configurar a identidade e o valor PSK com comprimento permitido para esses componentes.

Exceder os limites de comprimento resulta em falhas de comunicação entre o Zabbix componentes. :::

Antes que o servidor Zabbix se conecte ao agente usando o PSK, o servidor procura a identidade PSK e o valor PSK configurado para esse agente no banco de dados (na verdade no cache de configuração). Ao receber uma ligação, o agente usa a identidade PSK e o valor PSK de seu arquivo de configuração. Se ambos as partes têm a mesma cadeia de identidade PSK e valor PSK a conexão pode ter sucesso.

::: não importante Cada identidade PSK deve ser emparelhada com apenas um valor. É responsabilidade do usuário garantir que não haja dois PSKs com a mesma string de identidade, mas com valores diferentes. Falhando ao fazer isso pode levar a interrupções imprevisíveis de comunicação entre o Zabbix componentes usando PSKs com essa string de identidade PSK. :::

Gerando PSK

Por exemplo, um PSK de 256 bits (32 bytes) pode ser gerado usando o seguintes comandos:

- com OpenSSL:

```

$ openssl rand -hex 32
af8ced32dfe8714e548694e2d29e1a14ba6fa13f216cb35c19d0feb1084b0429

```

- com GnuTLS:

```
$ psktool -u psk_identity -p database.psk -s 32
Gerando uma chave aleatória para o usuário 'psk_identity'
Chave armazenada em database.psk
```

```
$ cat database.psk
psk_identity:9b8eafedfaae00cece62e85d5f4792c7d9c9bcc851b23216a1d300311cc4f7cb
```

Observe que "psktool" acima gera um arquivo de banco de dados com uma identidade PSK e seu PSK associado. O Zabbix espera apenas um PSK no arquivo PSK, então a string de identidade e os dois pontos (':') devem ser removidos do arquivo.

Configurando PSK para comunicação agente-servidor (exemplo)

No host do agente, grave o valor PSK em um arquivo, por exemplo, /home/zabbix/zabbix_agentd.psk. O arquivo deve conter PSK no primeiro cadeia de texto, por exemplo:

```
1f87b595725ac58dd977carne14b97461a7c1045b9a1c963065002c5473194952
```

Defina os direitos de acesso ao arquivo PSK - ele deve ser legível apenas pelo usuário do Zabbix.

Edite os parâmetros TLS no arquivo de configuração do agente zabbix_agentd.conf, por exemplo, defina:

```
TLSConnect=psk
TLSAccept=psk
TLSPSKFile=/home/zabbix/zabbix_agentd.psk
TLSPSKIdentity=PSK 001
```

O agente se conectará ao servidor (verificações ativas) e aceitará do servidor e zabbix_get somente conexões usando PSK. A identidade PSK será "PSK 001".

Reinicie o agente. Agora você pode testar a conexão usando zabbix_get, por exemplo:

```
$ zabbix_get -s 127.0.0.1 -k "system.cpu.load[all,avg1]" --tls-connect=psk \
--tls-psk-identity="PSK 001" --tls-psk-file=/home/zabbix/zabbix_agentd.psk
```

(Para minimizar o tempo de inatividade, veja como alterar o tipo de conexão em [Connection criptografia gerenciamento](#)).

Configure a criptografia PSK para este agente no frontend do Zabbix:

- Vá para: Configuração → Hosts
- Selecione o host e clique na guia **Criptografia**

Exemplo:

The screenshot shows the Zabbix web interface with the 'Encryption' tab selected. Under 'Connections to host', the 'PSK' button is highlighted. Under 'Connections from host', the 'PSK' checkbox is checked. The 'PSK identity' field is set to 'PSK 001'. The 'PSK' field contains a long alphanumeric string: '1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952'. At the bottom, there are buttons for 'Update', 'Clone', 'Full clone', 'Delete', and 'Cancel'.

Todos os campos de entrada obrigatórios estão marcados com um asterisco vermelho.

Quando o cache de configuração é sincronizado com o banco de dados, o novo conexões usarão PSK. Verifique se há erros nos arquivos de log do servidor e do agente mensagens.

Configurando PSK para servidor - comunicação proxy ativa (exemplo)

No proxy, grave o valor PSK em um arquivo, por exemplo, `/home/zabbix/zabbix_proxy.psk`. O arquivo deve conter PSK no primeiro cadeia de texto, por exemplo:

```
e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9
```

Defina os direitos de acesso ao arquivo PSK - ele deve ser legível apenas pelo usuário do Zabbix.

Edite os parâmetros TLS no arquivo de configuração de proxy `zabbix_proxy.conf`, para exemplo, defina:

```
TLSCConnect=psk
TLSPSKFile=/home/zabbix/zabbix_proxy.psk
TLSPSKIdentity=PSK 002
```

O proxy se conectará ao servidor usando o PSK. A identidade PSK será "PSK 002".

(Para minimizar o tempo de inatividade, veja como alterar o tipo de conexão em [Connection criptografia gerenciamento](#)).

Configure o PSK para este proxy no frontend do Zabbix. Vamos para Administração→Proxies, selecione o proxy, vá para a aba "Criptografia". Dentro "Conexões do proxy" marcam PSK. Cole no campo "Identidade PSK" "PSK 002" e "e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9" em campo "PSK". Clique em "Atualizar".

Reinicie o proxy. Ele começará a usar conexões criptografadas baseadas em PSK para servidor. Verifique os arquivos de log do servidor e do proxy para mensagens de erro.

Para um proxy passivo, o procedimento é muito semelhante. A única diferença - defina `TLSCAccept=psk` no arquivo de configuração do proxy e defina "Conexões para proxy" no frontend do Zabbix para PSK.

3 Solução de problemas

Recomendações gerais

- Comece entendendo qual componente atua como um cliente TLS e qual deles atua como um servidor TLS em caso de problema.
Servidor Zabbix, proxies e agentes, dependendo da interação entre eles, todos podem funcionar como servidores e clientes TLS.
Por exemplo, servidor Zabbix conectando-se ao agente para uma verificação passiva, atua como um cliente TLS. O agente está na função de servidor TLS.
O agente Zabbix, solicitando uma lista de verificações ativas do proxy, atua como um cliente TLS. O proxy está na função de servidor TLS.
Os utilitários `zabbix_get` e `zabbix_sender` sempre agem como TLS clientes.
- Zabbix usa autenticação mútua.
Cada lado verifica seu par e pode recusar a conexão.
Por exemplo, o servidor Zabbix conectado ao agente pode fechar a conexão imediatamente se o certificado do agente for inválido. E vice versa - O agente Zabbix aceitando uma conexão do servidor pode fechar a conexão se o servidor não for confiável para o agente.
- Examine os arquivos de log em ambos os lados - no cliente TLS e no servidor TLS.
O lado que recusa a conexão pode registrar uma razão precisa pela qual foi recusado. O outro lado geralmente relata um erro bastante geral (por exemplo, "Conexão fechada por peer", "conexão não foi corretamente encerrado").
- Às vezes, a criptografia mal configurada resulta em erro confuso mensagens de forma alguma apontando para uma causa real.
Nas subseções abaixo, tentamos fornecer um (longe de ser exaustivo) coleção de mensagens e possíveis causas que podem ajudar na solução de problemas.
Observe que diferentes kits de ferramentas de criptografia (OpenSSL, GnuTLS) geralmente produzir mensagens de erro diferentes nas mesmas situações de problema.
Às vezes, as mensagens de erro dependem até mesmo de uma combinação específica de kits de ferramentas de criptografia em ambos os lados.

1 Tipo de conexão ou problemas de permissão

O servidor está configurado para se conectar com o PSK ao agente, mas o agente aceita apenas conexões não criptografadas

No log do servidor ou proxy (com GnuTLS 3.3.16)

```
Falha ao obter valor do agente: zbx_tls_connect(): gnutls_handshake() falhou: \
-110 A conexão TLS foi finalizada incorretamente.
```

No log do servidor ou proxy (com OpenSSL 1.0.2c)

Falha ao obter valor do agente: conexão TCP bem-sucedida, não é possível estabelecer TLS para [[127.0.0.1

Conexão fechada por peer. Verifique os tipos de conexão permitidos e os direitos de acesso

Um lado se conecta com o certificado, mas o outro lado aceita apenas PSK ou vice-versa

Em qualquer log (com GnuTLS):

```
falhou ao aceitar uma conexão de entrada: de 127.0.0.1: zbx_tls_accept(): gnutls_handshake() falhou:\n-21 Não foi possível negociar um conjunto de cifras suportado.
```

Em qualquer log (com OpenSSL 1.0.2c):

```
falhou ao aceitar uma conexão de entrada: de 127.0.0.1: o handshake TLS retornou o código de erro 1:\narquivo .\ssl\s3_srvr.c linha 1411: erro:1408A0C1:rotinas SSL:ssl3_get_client_hello:nenhuma cifra com\nTLS escreve alerta fatal "falha de handshake"
```

Tentando usar o remetente Zabbix compilado com suporte a TLS para enviar dados para o servidor/proxy Zabbix compilado sem TLS

No log do lado da conexão:

Linux:

```
...Em zbx_tls_init_child()\n...Biblioteca OpenSSL (versão OpenSSL 1.1.1 11 de setembro de 2018) inicializada\n...\n...Em zbx_tls_connect(): psk_identity:"PSK teste remetente"\n...Fim do erro zbx_tls_connect():FAIL: 'conexão fechada por peer'\n...send value error: TCP bem-sucedido, não é possível estabelecer TLS para [[localhost]:10051]: conexão f
```

Janelas:

```
...Biblioteca OpenSSL (versão OpenSSL 1.1.1a 20 de novembro de 2018) inicializada\n...\n...Em zbx_tls_connect(): psk_identity:"PSK teste remetente"\n...zbx_psk_client_cb() solicitou a identidade PSK "PSK teste remetente"\n...Fim do erro zbx_tls_connect():FAIL: 'Erro de E/S SSL_connect(): [0x00000000] A operação foi concluída'\n...send value error: TCP bem-sucedido, não é possível estabelecer TLS para [[192.168.1.2]:10051]: SSL_con
```

No log do lado da aceitação:

```
... falhou ao aceitar uma conexão de entrada: de 127.0.0.1: suporte para TLS não foi compilado em
```

Um lado se conecta ao PSK, mas o outro lado usa LibreSSL ou foi compilado sem suporte de criptografia

LibreSSL não suporta PSK.

No log do lado da conexão:

```
...TCP bem-sucedido, não é possível estabelecer TLS para [[192.168.1.2]:10050]: SSL_connect() Erro de E/S
```

No log do lado da aceitação:

```
...falha ao aceitar uma conexão de entrada: de 192.168.1.2: suporte para PSK não foi compilado em
```

Na interface do Zabbix:

```
Falha ao obter valor do agente: TCP bem-sucedido, não é possível estabelecer TLS para [[192.168.1.2]:1005
```

Um lado se conecta ao PSK, mas o outro lado usa OpenSSL com suporte a PSK desativado

No log do lado da conexão:

```
...TCP bem-sucedido, não é possível estabelecer TLS para [[192.168.1.2]:10050]: SSL_connect() definir có
```

No log do lado da aceitação:

```
...falha ao aceitar uma conexão de entrada: de 192.168.1.2: código de resultado do conjunto de handshake
```

2 Problemas de certificado

OpenSSL usado com CRLs e para algumas CA na cadeia de certificados sua CRL não está incluída em TLSCRLFile

No log do servidor TLS no caso de OpenSSL peer:

falhou ao aceitar uma conexão de entrada: de 127.0.0.1: handshake TLS com 127.0.0.1 retornou o código de arquivo s3_srvr.c linha 3251: erro:14089086: rotinas SSL:ssl3_get_client_certificate:certificação de TLS escreve alerta fatal "CA desconhecida"

No log do servidor TLS no caso do par GnuTLS:

falhou ao aceitar uma conexão de entrada: de 127.0.0.1: handshake TLS com 127.0.0.1 retornou o código de arquivo rsa_pk1.c linha 103: erro:0407006A: rotinas rsa:RSA_padding_check_PKCS1_type_1:\n tipo de bloco não é 01 arquivo rsa_eay.c linha 705: erro:04067072: rotinas rsa:RSA_EAY_PUBLIC_DECRYPT

CRL expirou ou expira durante a operação do servidor

OpenSSL, no log do servidor:

- antes do vencimento:

não é possível conectar ao proxy "proxy-openssl-1.0.1e": TCP bem-sucedido, não é possível estabelecer TLS SSL_connect() retornou SSL_ERROR_SSL: arquivo s3_clnt.c linha 1253: erro:14090086:\n Rotinas SSL:ssl3_get_server_certificate:certificação de verificação falhou:\n TLS escreve alerta fatal "certificado revogado"

- após expiração:

não é possível conectar ao proxy "proxy-openssl-1.0.1e": TCP bem-sucedido, não é possível estabelecer TLS SSL_connect() retornou SSL_ERROR_SSL: arquivo s3_clnt.c linha 1253: erro:14090086:\n Rotinas SSL:ssl3_get_server_certificate:certificação de verificação falhou:\n Alerta fatal de gravação TLS "certificado expirado"

O ponto aqui é que, com CRL válida, um certificado revogado é relatado como "certificado revogado". Quando a CRL expira, a mensagem de erro muda para "certificado expirou", o que é bastante enganoso.

GnuTLS, no log do servidor:

- antes e depois do vencimento o mesmo:

não é possível conectar ao proxy "proxy-openssl-1.0.1e": TCP bem-sucedido, não é possível estabelecer TLS certificado de peer inválido: O certificado NÃO é confiável. A cadeia de certificados é revogada.

Certificado autoassinado, CA desconhecida

OpenSSL, no log:

error:'certificado autoassinado: SSL_connect() defina o código de resultado para SSL_ERROR_SSL: file ../s linha 1924: erro:1416F086:rotinas SSL:tls_process_server_certificate:certificate check failed:\n TLS escreve alerta fatal "CA desconhecida"

Isso foi observado quando o certificado do servidor por engano tinha o mesmo Emissor e Subject string, embora tenha sido assinado pela CA. Emissor e Sujeito são iguais no certificado CA de nível superior, mas não podem ser iguais no servidor certificado. (O mesmo se aplica a certificados de proxy e agente.)

3 problemas PSK

[comentário]: # (tags: ssl)

PSK contains an odd number of hex-digits

Proxy or agent does not start, message in the proxy or agent log:

invalid PSK in file "/home/zabbix/zabbix_proxy.psk"

PSK identity string longer than 128 bytes is passed to GnuTLS

In TLS client side log:

gnutls_handshake() failed: -110 The TLS connection was non-properly terminated.

In TLS server side log.

gnutls_handshake() failed: -90 The SRP username supplied is illegal.

Too long PSK value used with OpenSSL 1.1.1

In connecting-side log:

...OpenSSL library (version OpenSSL 1.1.1 11 Sep 2018) initialized

...

...In zbx_tls_connect(): psk_identity:"PSK 1"

```
...zbx_psk_client_cb() requested PSK identity "PSK 1"
...End of zbx_tls_connect():FAIL error:'SSL_connect() set result code to SSL_ERROR_SSL: file ssl\statem\ex
```

In accepting-side log:

```
...Message from 123.123.123.123 is missing header. Message ignored.
```

This problem typically arises when upgrading OpenSSL from 1.0.x or 1.1.0 to 1.1.1 and if the PSK value is longer than 512-bit (64-byte PSK, entered as 128 hexadecimal digits).

See also: [Value size limits](#)

18. Interface da Web

Visão geral Para um fácil acesso ao Zabbix de qualquer lugar e de qualquer plataforma, o interface baseada na web é fornecida.

Note:

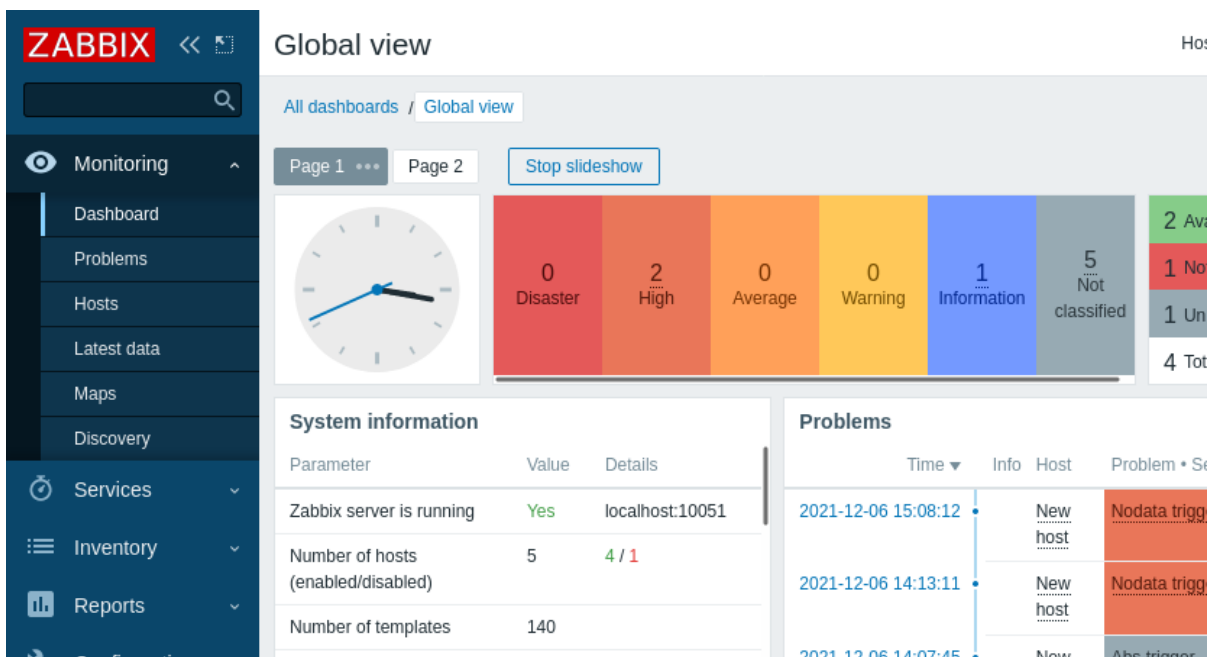
Se estiver usando mais de uma instância de front-end, certifique-se de que o localidades e bibliotecas (LDAP, SAML etc.) são instaladas e configuradas de forma idêntica para todos os frontends.

1 Menu

Visão geral

Um menu vertical em uma barra lateral fornece acesso a vários frontend do Zabbix Seções.

O menu é azul escuro no tema padrão.

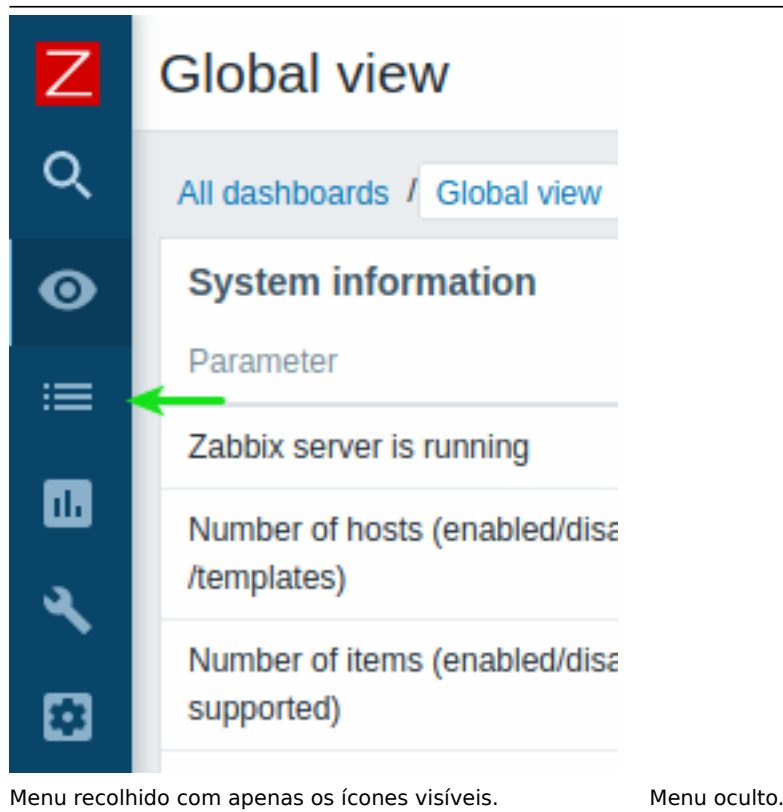


Trabalhando com o menu

Uma caixa de **pesquisa global** está localizada abaixo do logotipo do Zabbix.

O menu pode ser recolhido ou ocultado completamente:

- Para fechar, clique em  ao lado do logotipo do Zabbix
- Para ocultar, clique em  ao lado de Logo Zabbix



Menu recolhido

Quando o menu é reduzido apenas para ícones, um menu completo reaparece assim que como o cursor do mouse é colocado sobre ele. Observe que ele reaparece na página contendo; para mover o conteúdo da página para a direita, você deve clicar no botão expandir. Se o cursor do mouse for colocado novamente fora do menu, o menu entrará em colapso novamente após dois segundos.

Você também pode fazer um menu recolhido reaparecer totalmente pressionando a guia chave. Pressionar a tecla Tab repetidamente permitirá focar no próximo menu elemento.

Menu oculto

Mesmo quando o menu está completamente oculto, um menu completo é apenas um mouse clique de distância, clicando no ícone do hambúrguer. Observe que ele reaparece o conteúdo da página; para mover o conteúdo da página para a direita, você precisa exibir o menu clicando no botão Mostrar barra lateral.

2 Seções de Frontend

Por favor, utilize a barra lateral para acessar o conteúdo na seção de "Frontend sections".

1 Monitoramento


Visão geral

O menu Monitoramento tem tudo a ver com a exibição de dados. Qualquer informação O Zabbix está configurado para coletar, visualizar e agir, será exibido nas várias seções do menu Monitoramento.

Botões do modo de visualização

Os seguintes botões localizados no canto superior direito são comuns para cada seção:

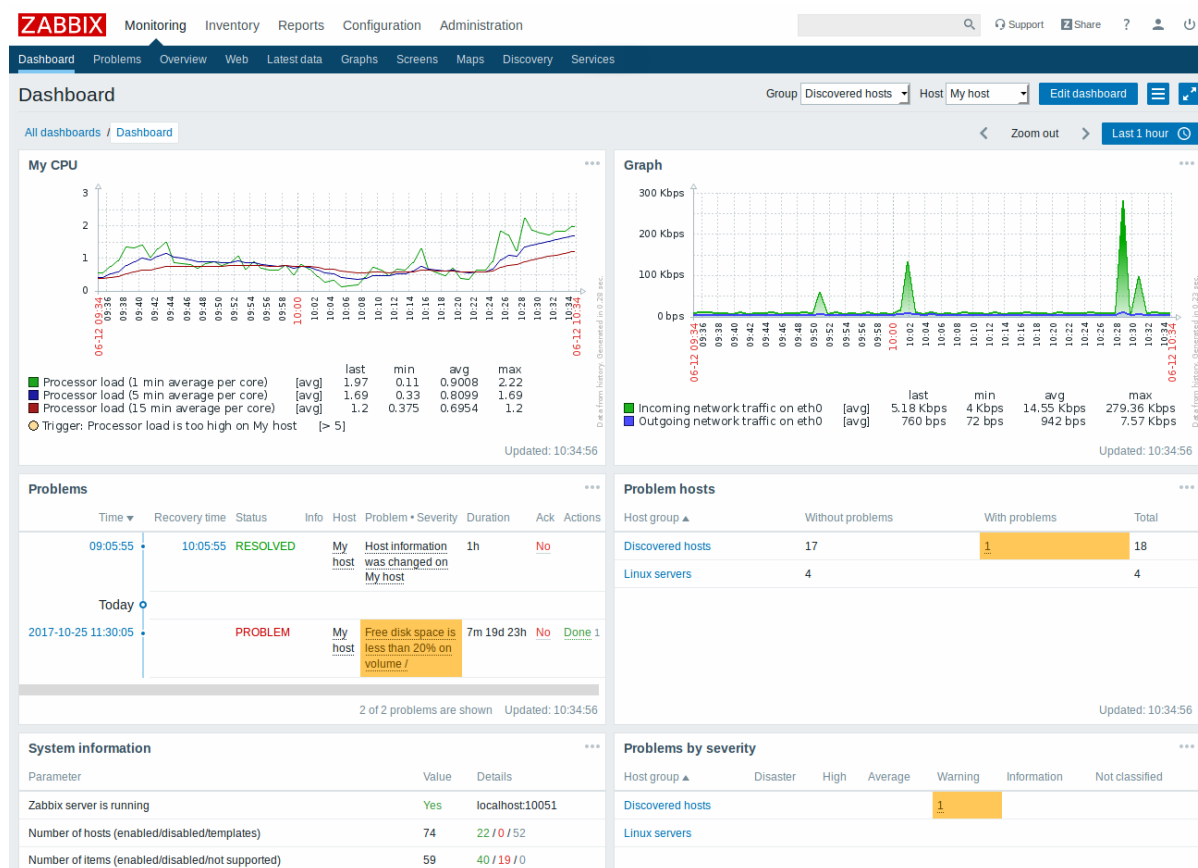


Exibir página no modo quiosque. Nesse modo, apenas o conteúdo da página é exibido.
Para sair do modo quiosque, mova o cursor do mouse até o  botão de saída aparece e clique nele. Você será levado de volta ao modo normal.

1 Dashboard

Visão geral

O módulo Monitoramento → Dashboard é similar a um painel de um carro, apresentando as informações mais importantes.



Os widgets do Dashboard podem ser agrupados em dois tipos: Favoritos e de Status. Os favoritos tem por função possibilitar a adição de links para acesso fácil, enquanto os de status tem por função mostrar visões agrupadas de determinado tipo de informação.

Todos eles suportam customização, podendo ser expandidos ou contraídos e ter sua posição modificada no dashboard.

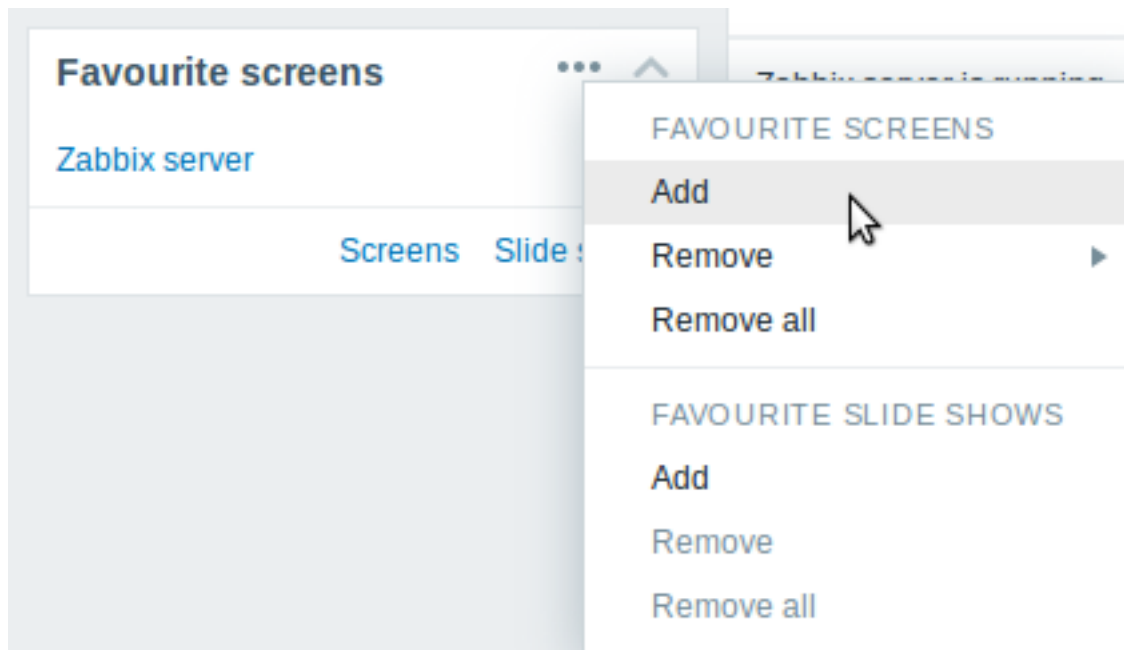
Widgets de favoritos

Temos aqui alguns widgets para itens favoritos que possibilitam a criação de atalhos rápidos para os gráficos, gráficos customizados, apresentações de slides e mapas que você mais usa.

Para adicionar itens a um widget (exemplo para telas):

- Clique no botão menu do widget (botão com reticências)
- Selecione a opção Adicionar do menu flutuante
- Selecione no menu pop-up a(s) tela(s) desejada(s)
- Clique no botão Selecionar

Agora seu widget de telas favoritas tem novos atalhos para acesso fácil às suas telas.



Widgets de status

Existem vários widgets de status:

- Status do Zabbix - Mostra uma visão agrupada do ambiente Zabbix
- Status do sistema - Mostra uma visão agrupada dos incidentes classificados por severidade e grupo
- Status dos hosts - Mostra uma visão agrupada dos incidentes classificados grupo e ocorrência ou não de incidentes ativos
- Últimos 20 incidentes - Mostra os últimos 20 incidentes
- Monitoramento Web - Resumo das monitorações web agrupado por regra e resultado (sucesso / falha)
- Status das regras de autobusca - Resumo das regras de autobusca ativas (se não existir nenhuma regra este widget não aparece)

Dashboard filter



Clique no ícone na barra de título para ter acesso ao controle de filtros do dashboard.

Dashboard

Dashboard filter **Enabled**

Host groups **Selected**

Show selected groups **Select**

Hide selected groups **Select**

Hosts ☒ Show hosts in maintenance

Triggers with severity

- ☐ Not classified
- ☐ Information
- ☒ Warning
- ☒ Average
- ☒ High
- ☒ Disaster

Trigger name

Problem display **All** **Separated** **Unacknowledged only**

Update

Cancel

Ao ativar o filtro você pode limitar quais hosts e triggers serão apresentadas no dashboard e definir como os incidentes serão apresentados.

Parâmetro	Descrição
Filtro de dashboard	Marque para ativar o filtro.
Grupos de hosts	Selecione para mostrar dados de: Todos - todos os grupos de hosts Selecionados - apenas dos grupos selecionados.
Exibir grupos selecionados	Este campo fica disponível quando o campo Grupos de hosts está com a opção Selecionados marcada. Entre com os grupos de hosts para apresentar, este campo possui recurso de auto-completar. Apenas os dados de grupos selecionados estarão visíveis no dashboard, se nenhum for informado todos serão exibidos.
Ocultar grupos selecionados	Este campo fica disponível quando o campo Grupos de hosts está com a opção Selecionados marcada. Entre com os grupos de hosts para NÃO EXIBIR, este campo possui recurso de auto-completar. Os grupos de hosts selecionados aqui não estarão visíveis no dashboard. É importante salientar que a ocultação de um grupo pode afetar a exibição de hosts em outros grupos, exemplo: Se os hosts 001, 002, 003 estão no grupo A e os hosts 002, 003 também estão no grupo B. E nós selecionamos para Exibir o grupo A e Esconder o grupo B ao mesmo tempo, apenas os dados do host 001 serão apresentados no dashboard.

Parâmetro	Descrição
Hosts	Selecione a opção Mostrar hosts em manutenção para apresentar dados de hosts que estão sob efeito de regras de manutenção programada no Dashboard.
Triggers com severidade	Marque as severidades de triggers que você deseja exibir no dashboard.
Nome de trigger como	Limite a apresentação de triggers nos widgets: Status do sistema, Status dos Hosts e Últimos 20 incidentes ao filtrar pelo nome das triggers.
Exibição de incidentes	Contabilizar os incidentes pela tratativa: Todos - Considera todos os incidentes Separados - incidentes não reconhecidos são contabilizados em separado do total de incidentes Somente não reconhecidos - apenas os incidentes não reconhecidos serão contabilizados.

Se o filtro Dashboard estiver ativo, o ícone de ajuste de filtro terá um indicador verde para indicar a existência do filtro ativo:



Dashboard size

The minimum width of a dashboard is 1200 pixels. The dashboard will not shrink below this width; instead a horizontal scrollbar is displayed if the browser window is smaller than that.

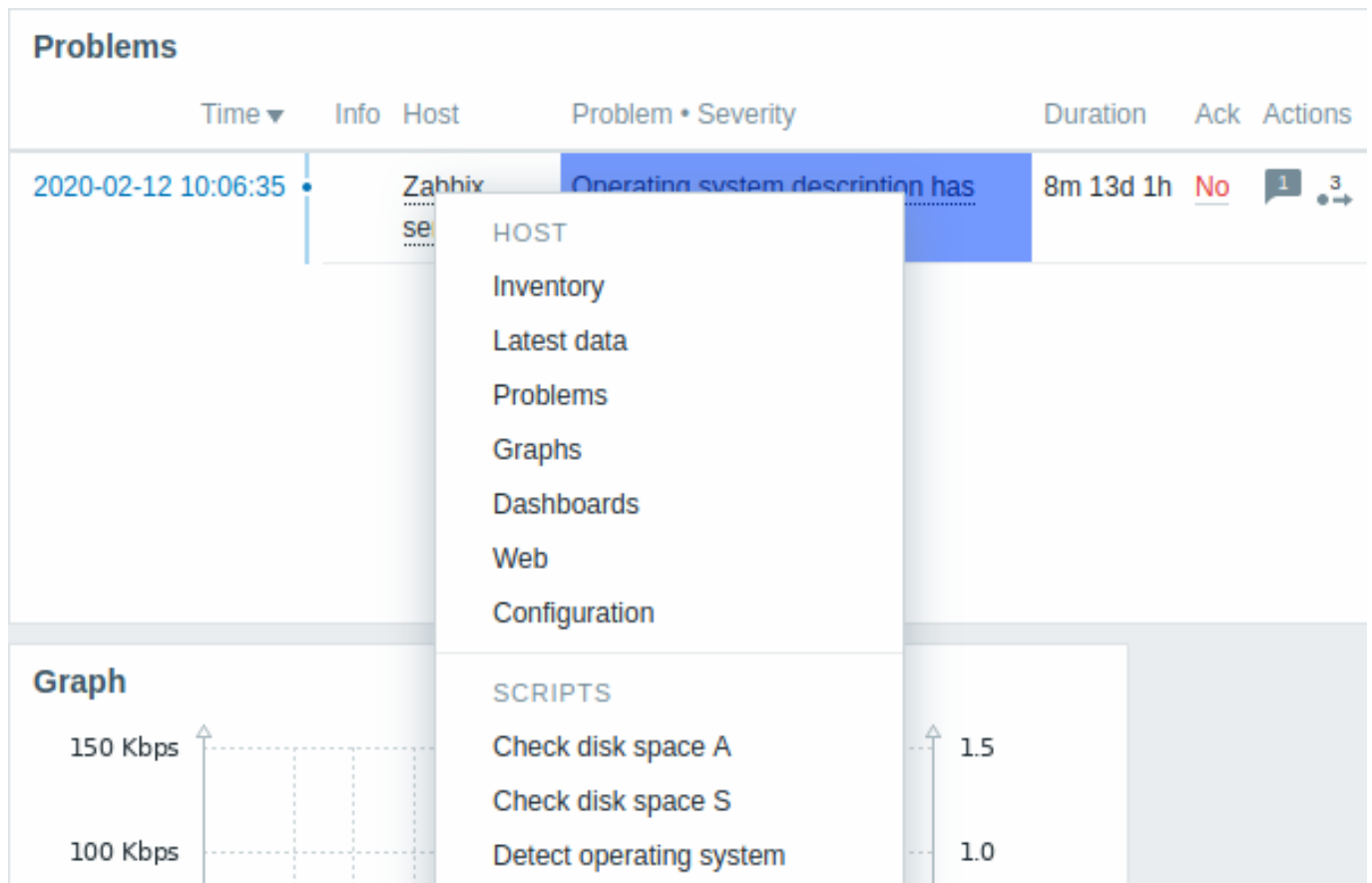
The maximum width of a dashboard is the browser window width. Dashboard widgets stretch horizontally to fit the window. At the same time, a dashboard widget cannot be stretched horizontally beyond the window limits.

Technically the dashboard consists of 12 horizontal columns of always equal width that stretch/shrink dynamically (but not to less than 1200 pixels total).

Vertically the dashboard may contain a maximum of 64 rows. Each row has a fixed height of 70 pixels. A widget may be up to 32 rows high.

Menu flutuante host

Ao clicar em um host no widget dos Últimos 20 incidentes será apresentado o menu flutuante do host. Este menu inclui links para scripts customizados, dados recentes, triggers, inventário, gráficos e telas daquele host.

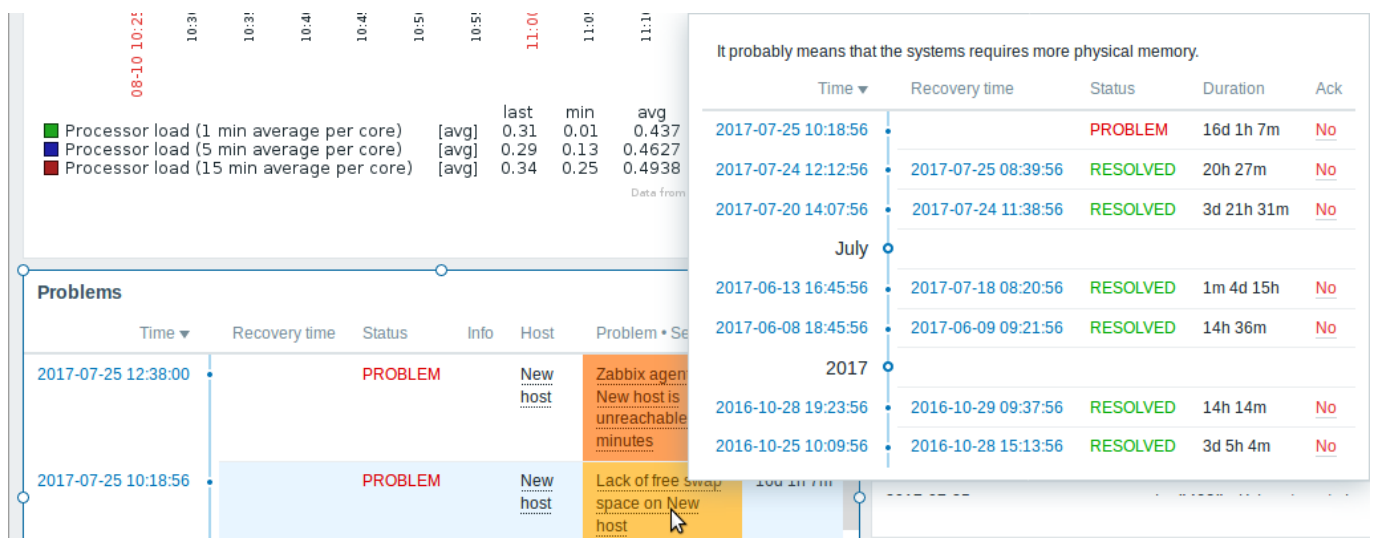


O menu de host é acessível ao clicar em um host em diversos locais:

- Monitoramento → **Visão geral** (na célula do host)
- Monitoramento → **Dados recentes** (ao clicar no nome do host)
- Monitoramento → **Triggers** (ao clicar no nome do host)
- Monitoramento → **Eventos** (ao clicar no nome do host)
- Monitoramento → **Eventos** → Detalhes do evento (ao clicar no nome do host)
- Monitoramento → **Telas** (nos widgets de Incidentes do Host e Incidentes do grupo de host)
- Monitoramento → **Mapas** (ao clicar no ícone associado ao host)
- Relatórios → **Top 100 de triggers**

Menu flutuante de triggers

Clicando no Incidente apresentado no widget dos Últimos 20 incidentes será apresentado o menu flutuante de triggers e eventos. Ele contém a lista de eventos e, se definido, a descrição da trigger e uma URL clicável.



Creating a dashboard

It is possible to create a new dashboard in two ways:

- Click on Create dashboard, when viewing all dashboards
- Select Create new from the action menu, when viewing a single dashboard

You will be first asked to enter general dashboard parameters:

Dashboard properties

*

Owner

Admin (Zabbix Administrator)

×

Select

*

Name

New dashboard

Default page display period

30 seconds

▼

Start slideshow automatically

☒

Apply

Cancel

Parameter	Description
Owner	Select system user that will be the dashboard owner.
Name	Enter dashboard name.
Default page display period	Select period for how long a dashboard page is displayed before rotating to the next page in a slideshow .
Start slideshow automatically	Mark this checkbox to run a slideshow automatically one more than one dashboard page exists.

When you click on Apply, an empty dashboard is opened:

≡

New dashboard

⚙

+ Add

▼

Save changes

Cancel

All dashboards / New dashboard

Page 1

...

+

+

Add a new widget

To populate the dashboard, you can add widgets and pages.

Click on the Save changes button to save the dashboard. If you click on Cancel, the dashboard will not be created.

Adicionando widgets

Para adicionar um widget a um painel:

- Clique no

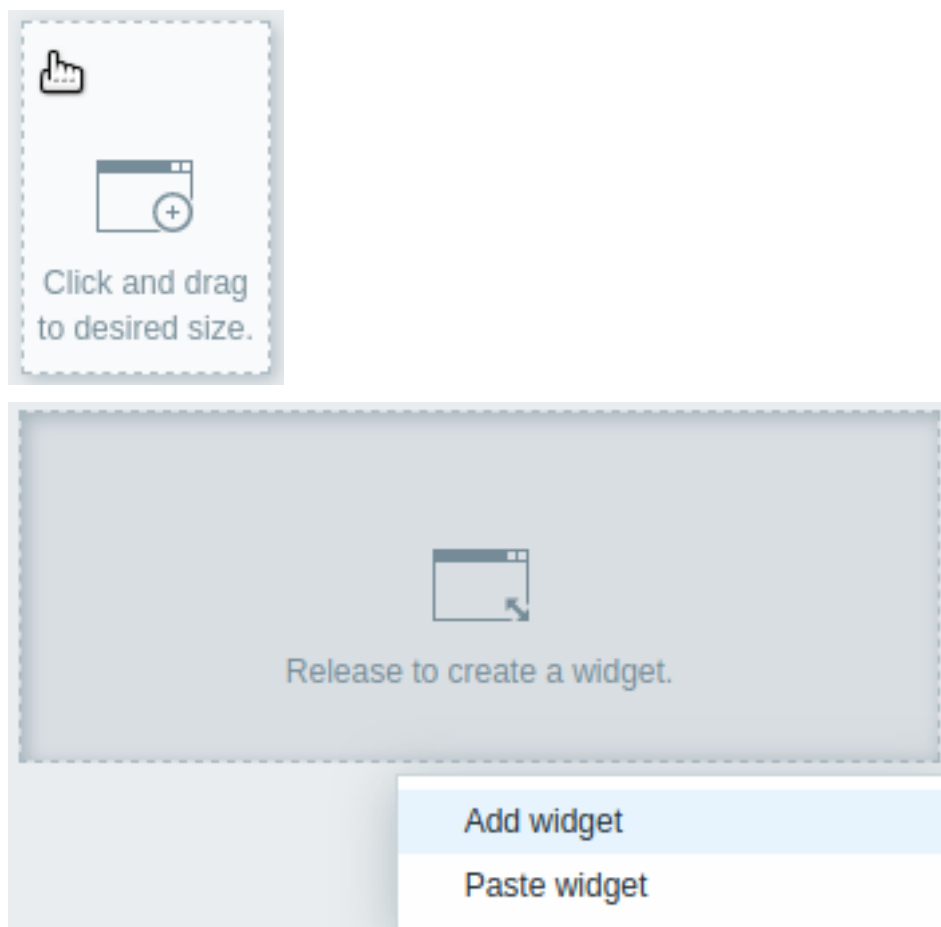
+ Add

 botão ou a opção Adicionar widget no menu de ação que pode ser aberto clicando na seta. Preencha o formulário de configuração do widget. O widget será criado em seu tamanho padrão e colocado após o widgets existentes (se houver);

Ou

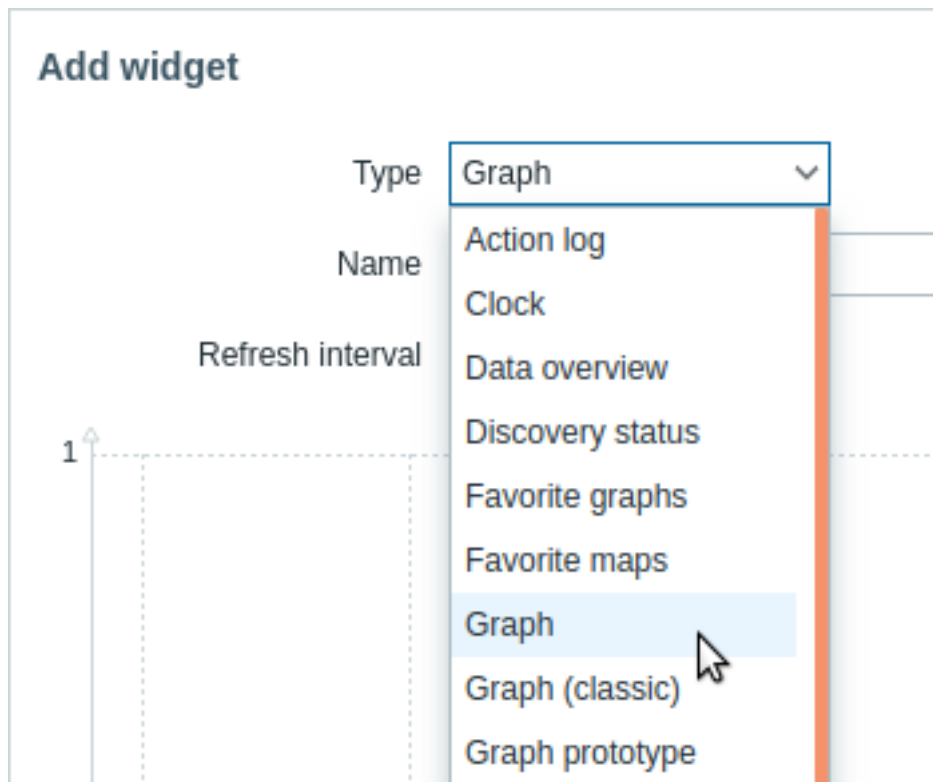
- Mova o mouse para o local vazio desejado para o novo widget. Perceber como um espaço reservado aparece, ao passar o mouse, em qualquer slot vazio no painel de controle. Em seguida, clique para abrir o formulário de configuração do widget. Depois de preenchendo o formulário o widget será criado em seu tamanho padrão ou, se o tamanho padrão for maior do que

o disponível, pegue o espaço disponível. Alternativamente, você pode clicar e arrastar o espaço reservado para o tamanho de widget desejado, solte e preencha o formulário de configuração do widget. (Observe que quando há um widget copiado para a área de transferência, primeiro você será solicitado a selecionar entre as opções Adicionar widget e Colar widget para criar um widget.)



No formulário de configuração do widget:

- Selecione o Tipo de widget
- Insira os parâmetros do widget
- Clique em Adicionar





Widgets

Os seguintes widgets podem ser adicionados a um painel:

- Ação log
- Relógio
- Dados visão geral
- Descoberta status
- Favorito gráficos
- Favorito mapas
- Geomap
- Gráfico
- Gráfico (clássico)
- Gráfico protótipo
- Hospedeiro disponibilidade
- Valor do item
- Mapa
- Navegação no mapa tree
- Simples text
- Problema hosts
- Problemas
- Sistema informações
- Problemas por gravidade
- Principais hosts
- Acionar visão geral
- URL
- Rede monitoramento

No modo de edição do painel, os widgets podem ser redimensionados e movidos painel clicando na barra de título do widget e arrastando-o para um novo localização. Além disso, você pode clicar nos seguintes botões no canto superior direito canto do widget para:

-  - editar um widget;
-  - acesse o **menu widget**

Clique em Salvar alterações para que o painel faça as alterações no widgets permanentes.

Copying/pasting widgets

Dashboard widgets can be copied and pasted, allowing to create a new widget with the properties of an existing one. They can be copy-pasted within the same dashboard, or between dashboards opened in different tabs.

A widget can be copied using the **widget menu**. To paste the widget:

- click on the arrow next to the Add button and selecting the Paste widget option, when editing the dashboard
- use the Paste widget option when adding a new widget by selecting some area in the dashboard (a widget must be copied first for the paste option to become available)

A copied widget can be used to paste over an existing widget using the Paste option in the **widget menu**.

Creating a slideshow

A slideshow will run automatically if the dashboard contains two or more pages (see **Adding pages**) and if one of the following is true:





- The Start slideshow automatically option is marked in dashboard properties
- The dashboard URL contains a `slideshow=1` parameter

The pages rotate according to the intervals given in the properties of the dashboard and individual pages. Click on:

- Stop slideshow - to stop the slideshow
- Start slideshow - to start the slideshow



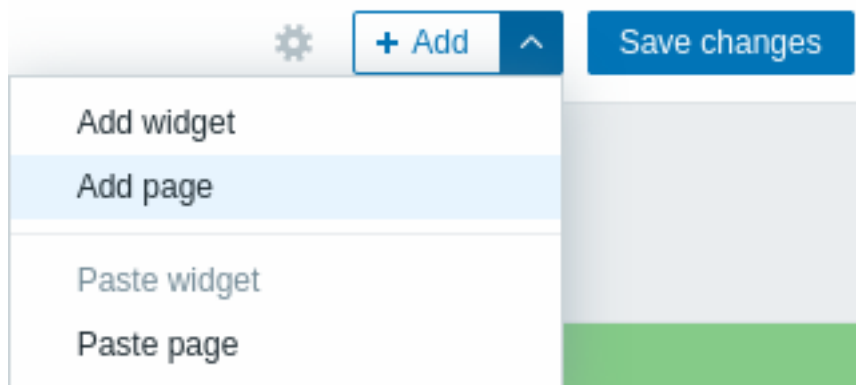
Slideshow-related controls are also available in **kiosk mode** (where only the page content is shown):

-  - stop slideshow
-  - start slideshow
-  - go back one page
-  - go to the next page

Adding pages

To add a new page to a dashboard:

- Make sure the dashboard is in the **editing mode**
- Click on the arrow next to the Add button and select the Add page option



- Fill the general page parameters and click on Apply. If you leave the name empty, the page will be added with a Page N name where 'N' is the incremental number of the page. The page display period allows to customize how long a page is displayed in a slideshow.

Dashboard page properties

Name

Page 2

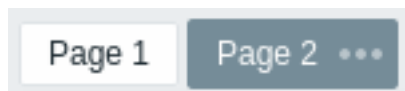
Page display period

Default (30 seconds) ▾

Apply

Cancel

A new page will be added, indicated by a new tab (Page 2).



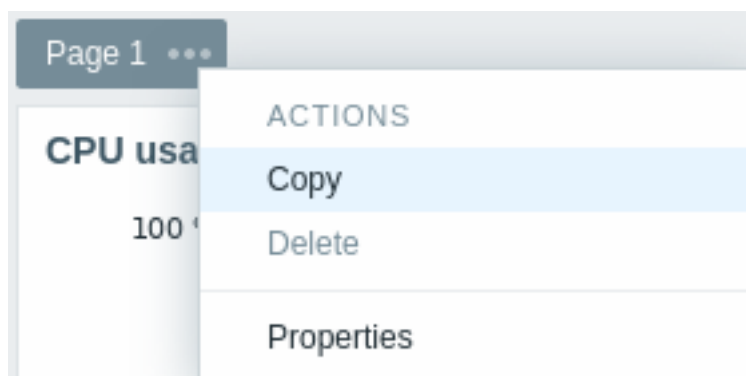
The pages can be reordered by dragging-and-dropping the page tabs. Reordering maintains the original page naming. It is always possible to go to each page by clicking on its tab.

When a new page is added, it is empty. You can add widgets to it as described above.

Copying/pasting pages

Dashboard pages can be copied and pasted, allowing to create a new page with the properties of an existing one. They can be pasted from the same dashboard or a different dashboard.


To paste an existing page to the dashboard, first copy it, using the **page menu**:

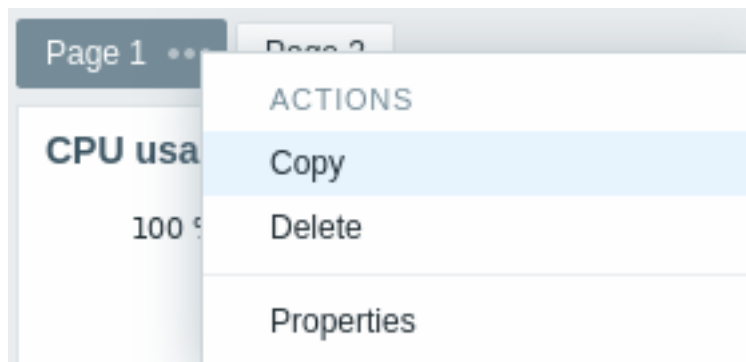


To paste the copied page:

- Make sure the dashboard is in the **editing mode**
- Click on the arrow next to the Add button and select the Paste page option

Page menu

The page menu can be opened by clicking on the three dots  next to the page name:



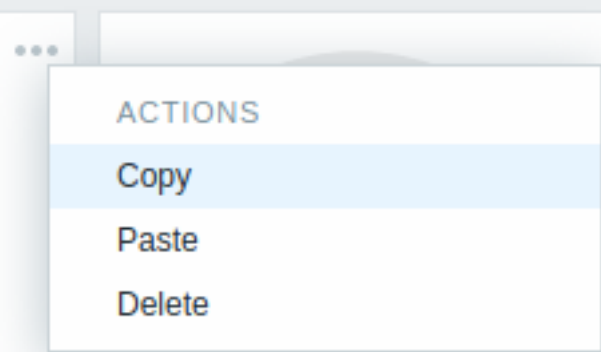
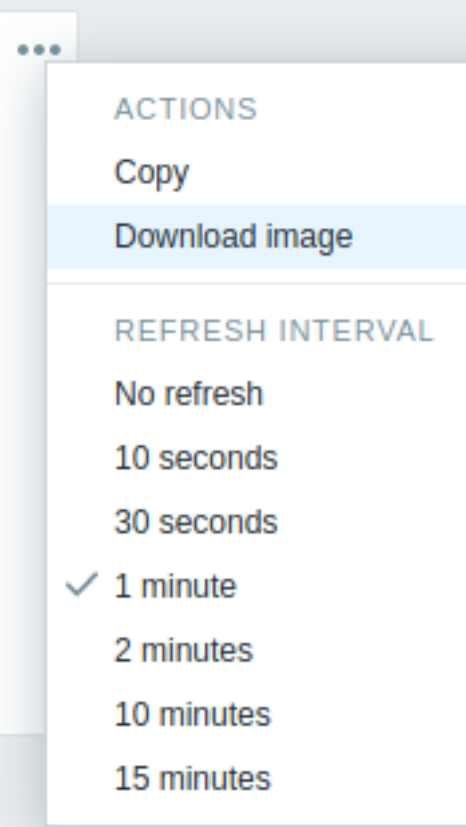
It contains the following options:

- Copy - copy the page
- Delete - delete the page (pages can only be deleted in the dashboard editing mode)

- Properties - customize the page parameters (the name and the page display period in a slideshow)

Widget menu

The widget menu contains different options based on whether the dashboard is in the edit or view mode:

Widget menu	Options
<p>In dashboard edit mode:</p> 	<p>Copy - copy the widget</p> <p>Paste - paste a copied widget over this widget This option is grayed out if no widget has been copied.</p> <p>Delete - delete the widget</p>
<p>In dashboard view mode:</p> 	<p>Copy - copy the widget</p> <p>Download image - download the widget as a PNG image (only available for graph/classic graph widgets)</p> <p>Refresh interval - select the frequency of refreshing the widget contents</p>

Dynamic widgets

When **configuring** some of the widgets:

- Graphs (simple and custom)
- Plain text
- URL

there is an extra option called Dynamic item. You can check this box to make the widget dynamic - i.e. capable of displaying different content based on the selected host.

Now, when saving the dashboard, you will notice that a new host selection field has appeared atop the dashboard for selecting the host (while the Select button allows selecting the host group in a popup):

Host

Thus you have a widget, which can display content that is based on the data from the host that is selected. The benefit of this is that you do not need to create extra widgets just because, for example, you want to see the same graphs containing data from various hosts.

Permissions to dashboards

Permissions to dashboards for regular users and users of 'Admin' type are limited in the following way:

- They can see and clone a dashboard if they have at least READ rights to it;
- They can edit and delete dashboard only if they have READ/WRITE rights to it;
- They cannot change the dashboard owner.

Host menu

Clicking on a host in the Problems widget brings up the host menu. It includes links to host inventory, latest data, problems, graphs, dashboards, web scenarios and configuration. Note that host configuration is available for Admin and Superadmin users only.

The screenshot shows the Zabbix Problems widget. At the top, there's a table with columns: Time, Info, Host, Problem • Severity, Duration, Ack, and Actions. A row is selected for the host 'Zabbix' with the problem 'Operating system description has'. A dropdown menu is open for the 'Host' column, showing options: HOST, Inventory, Latest data, Problems, Graphs, Dashboards, Web, and Configuration. Below the table, there's a 'Graph' section showing a line graph for '150 Kbps' and '100 Kbps'. To the right of the graph, there's a 'SCRIPTS' section with a list of scripts: 'Check disk space A', 'Check disk space S', and 'Detect operating system'.

Global scripts can also be run from the host menu. These scripts need to have their scope defined as 'Manual host action' to be available in the host menu.

The host menu is accessible by clicking on a host in several other frontend sections:

- Monitoring → Problems
- Monitoring → Problems → Event details
- Monitoring → Hosts
- Monitoring → Hosts → Web Monitoring
- Monitoring → Latest data
- Monitoring → Maps
- Reports → Triggers top 100

Problem event popup

The problem event popup includes the list of problem events for this trigger and, if defined, the trigger description and a clickable URL.

The screenshot shows the Zabbix Problems widget with a problem event popup. The popup displays a list of problem events for the trigger 'Disk space is critically low (>90% used)'. The events are listed in a table with columns: Time, Info, Host, Problem • Severity, Duration, Ack, and Actions. The events are sorted by time, with the most recent event at the top. The popup also shows the trigger description and a clickable URL.

To bring up the problem event popup:

- roll a mouse over the problem duration in the Duration column of the Problems widget. The popup disappears once you remove the mouse from the duration.
- click on the duration in the Duration column of the Problems widget. The popup disappears only if you click on the duration again.

Attention:

Resolved values of {ITEM.VALUE} and {ITEM.LASTVALUE} macros in trigger descriptions are truncated to 20 characters. To see the entire values you may use **macro functions** with these macros, e.g. `{{ITEM.VALUE}.regsub("(.*)", \1)}`, `{{ITEM.LASTVALUE}.regsub("(.*)", \1)}` as a workaround.

1 Dashboard widgets

Visão geral


Esta seção fornece os detalhes dos parâmetros que são comuns para **dashboard** widgets.

Parâmetros comuns

Os seguintes parâmetros são comuns para cada widget:

Nome	Digite um nome de widget.
Intervalo de atualização	Configurar intervalo de atualização padrão. Os intervalos de atualização padrão para widgets variam de Sem atualização a 15 minutos dependendo do tipo de widget. Por exemplo: Sem atualização para widget de URL, 1 minuto para widget de log de ação, 15 minutos para widget de relógio.
Mostrar cabeçalho	Marque a caixa de seleção para mostrar o cabeçalho permanentemente. Quando desmarcado, o cabeçalho fica oculto para economizar espaço e só desliza para cima e fica visível novamente quando o mouse é posicionado sobre o widget, tanto nos modos de visualização quanto de edição. Também é semi-visível ao arrastar um widget para um novo local.

Os intervalos de atualização de um widget podem ser definidos como um valor padrão para todos os usuários correspondentes e também cada usuário pode definir seu próprio intervalo de atualização valor:

- Para definir um valor padrão para todos os usuários correspondentes, alterne para modo de edição (clique no botão Editar painel, encontre o widget, clique no botão Editar abrindo o formulário de edição de um widget) e escolha o intervalo de atualização necessário no menu suspenso Lista.
- Definir um intervalo de atualização exclusivo para cada usuário separadamente é possível no modo de visualização clicando no  botão para um determinado widget.

O intervalo de atualização exclusivo definido por um usuário tem prioridade sobre o widget configuração e, uma vez definido, é sempre preservado quando o widget configuração é modificada.

Para ver **parâmetros específicos** de cada widget, acesse as páginas de widgets individuais para:

- Registro de ações
- Relógio
- Status de descoberta
- Gráficos favoritos
- Mapas favoritos
- Geomap
- Gráfico
- Gráfico (clássico)
- Protótipo de gráfico
- Disponibilidade do host
- Valor do item
- Mapa
- Árvore de navegação do mapa

- Texto simples
- Hosts com problemas
- Problemas
- relatório de SLA
- Informações do sistema
- Problemas por gravidade
- Principais hosts
- Visão geral do gatilho
- URL
- Monitoramento da Web

Widgets obsoletos:

- Visão geral dos dados

:::não importante Widgets obsoletos serão removidos na próxima versão principal. :::

1 Registro de ação

Visão geral

No widget de log de ação, você pode exibir detalhes das operações de ação (notificações, comandos remotos). Ele replica informações de Administração → Auditoria.

Configuração

Para configurar, selecione Registro de ações como tipo:

Add widget

Type
Action log
Show header
☒

Name
Action log

Refresh interval
Default (1 minute)

Sort entries by
Time (descending)

* Show lines
25

Add
Cancel

Além dos parâmetros que são comuns para todos os widgets, você pode definir as seguintes opções específicas:

Classificar entradas por

Classificar entradas por:

Hora (decrecente ou crescente)

Tipo (decrecente ou crescente)

Status (decrecente ou crescente)

Destinatário (descendente ou ascendente).

Mostrar linhas

Defina quantas linhas de log de ação serão exibidas no widget.

2 Relógio

Visão geral

No widget de relógio, você pode exibir local, servidor ou host especificado Tempo.

Configuração

Para configurar, selecione Relógio como tipo:

Add widget

Type

Clock

Show header

☒

Name

Local time

Refresh interval

Default (15 minutes)

Time type

Local time

Add

Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Tipo de horário	Selecione o horário local, servidor ou host especificado. O horário do servidor será idêntico ao fuso horário definido globalmente ou para o usuário do Zabbix.
Item	Selecione o item para exibir o tempo. Para exibir a hora do host, use o item <code>system.localtime[local]</code> . Este item deve existir no host. Este campo está disponível somente quando Horário do host é selecionado.

3 Data overview

:::não importante Este widget está obsoleto e será removido na próxima versão principal. :::

Overview

In the data overview widget, you can display the latest data for a group of hosts.

The color of problem items is based on the problem severity color, which can be adjusted in the **problem update** screen.

By default, only values that fall within the last 24 hours are displayed. This limit has been introduced with the aim of improving initial loading times for large pages of latest data. This limit is configurable in Administration → General → **GUI**, using the Max history display period option.

Clicking on a piece of data offers links to some predefined graphs or latest values.

Note that 50 records are displayed by default (configurable in Administration → General → **GUI**, using the Max number of columns and rows in overview tables option). If more records exist than are configured to display, a message is displayed at the bottom of the table, asking to provide more specific filtering criteria. There is no pagination.

Configuration

To configure, select Data overview as type:

Add widget
✕

Type
Data overview

Show header
☒

Name
default

Refresh interval
Default (1 minute)

Host groups
type here to search
Select

Hosts
type here to search
Select

Tags

And/Or
Or

tag
Contains
value
Remove

Add

Show suppressed problems
☐

Hosts location

Left
Top

Add
Cancel

In addition to the parameters that are **common** for all widgets, you may set the following specific options:

Host groups	Select host groups. This field is auto-complete so starting to type the name of a group will offer a dropdown of matching groups. Scroll down to select. Click on 'x' to remove the selected.
Hosts	Select hosts. This field is auto-complete so starting to type the name of a host will offer a dropdown of matching hosts. Scroll down to select. Click on 'x' to remove the selected.
Tags	Specify tags to limit the number of item data displayed in the widget. It is possible to include as well as exclude specific tags and tag values. Several conditions can be set. Tag name matching is always case-sensitive. There are several operators available for each condition: Exists - include the specified tag names Equals - include the specified tag names and values (case-sensitive) Contains - include the specified tag names where the tag values contain the entered string (substring match, case-insensitive) Does not exist - exclude the specified tag names Does not equal - exclude the specified tag names and values (case-sensitive) Does not contain - exclude the specified tag names where the tag values contain the entered string (substring match, case-insensitive) There are two calculation types for conditions: And/Or - all conditions must be met, conditions having the same tag name will be grouped by the Or condition Or - enough if one condition is met
Show suppressed problems	Mark the checkbox to display problems that would otherwise be suppressed (not shown) because of host maintenance.

4 Status de descoberta

Visão geral

Este widget exibe um resumo do status da descoberta de rede ativa as regras.

Add widget ✕

Type

Discovery status ▼

Name

Discovery status

Refresh interval

Default (1 minute) ▼

Show header

☒

Add


Cancel

Todos os parâmetros de configuração são **comuns** para todos os widgets.

5 gráficos favoritos

Visão geral

Este widget contém atalhos para os gráficos mais necessários, ordenados alfabeticamente.


A lista de atalhos é preenchida quando você **visualizar** um gráfico e, em seguida, clique em seu  Botão Adicionar aos favoritos.

Todos os parâmetros de configuração são **comuns** para todos os widgets.

6 mapas favoritos

Visão geral

Este widget contém atalhos para os mapas mais necessários, ordenados alfabeticamente.

A lista de atalhos é preenchida quando você **visualizar** um mapa e, em seguida, clique em seu  Botão Adicionar aos favoritos.

Todos os parâmetros de configuração são **comuns** para todos os widgets.

7 Geomapa

[comentário]: # (tags: mapa)

Visão geral

O widget Geomap exibe hosts como marcadores em um mapa geográfico usando Biblioteca de mapas interativos JavaScript de código aberto Leaflet.

Note:

O Zabbix oferece vários serviços de mosaicos de mapas predefinidos provedores e uma opção para adicionar um provedor de serviço de tile personalizado ou até mesmo próprios blocos de host (configuráveis em Administração → Geral → Mapas geográficos **menu seção**).

Por padrão, o widget exibe todos os hosts habilitados com coordenadas geográficas válidas definidas na configuração do host. Isto é possível configurar a filtragem de host nos parâmetros do widget.

As coordenadas de host válidas são:

- Latitude: de -90 a 90 (pode ser inteiro ou float)
- Longitude: de -180 a 180 (pode ser inteiro ou float)

Configuração

Para adicionar o widget, selecione Geomap como tipo.

The 'Add widget' dialog box for Geomap includes the following fields and controls:

- Type:** A dropdown menu set to 'Geomap'.
- Show header:** A checkbox that is checked.
- Name:** A text input field containing 'default'.
- Refresh interval:** A dropdown menu set to 'Default (1 minute)'.
- Host groups:** A text input field with the placeholder 'type here to search' and a 'Select' button.
- Hosts:** A text input field with the placeholder 'type here to search' and a 'Select' button.
- Tags:** Two buttons labeled 'And/Or' and 'Or'.
- Filter:** A section with a text input 'tag', a dropdown menu set to 'Contains', a text input 'value', and a 'Remove' button.
- Add:** A blue button below the filter section.
- Initial view:** A text input field containing '40.6892494,-74.0466891' with a help icon (?) to its left.
- Buttons:** 'Add' and 'Cancel' buttons at the bottom right.

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Grupos de hosts

Selecione os grupos de hosts a serem exibidos no mapa. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes. Role para baixo para selecionar. Clique em 'x' para remover os grupos selecionados.

Hosts

Se nada for selecionado nos campos Host groups e Hosts, todos os hosts com coordenadas válidas serão exibidos. Selecione os hosts a serem exibidos em todo o mapa. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um host oferecerá uma lista suspensa de hosts correspondentes. Role para baixo para selecionar. Clique em 'x' para remover hosts selecionados. Se nada for selecionado nos campos Host groups e Hosts, todos os hosts com coordenadas válidas serão exibidos.

Tags

Especifique as tags para limitar o número de hosts exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.

Há vários operadores disponíveis para cada condição:

Exists - inclua os nomes de tags especificados

Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)

Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)

Não existe - exclui o especificado nomes de tag

Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)

Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)

Existem dois tipos de cálculo para condições:

E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição

Ou - suficiente se uma condição for atendida

Coordenadas centrais separadas por vírgulas e um nível de zoom opcional para exibir quando o widget é carregado inicialmente no formato

<latitude>,<longitude>,<zoom>

Se o zoom inicial for especificado, o O widget de mapa geográfico é carregado no nível de zoom fornecido. Caso contrário, o zoom inicial é calculado como metade do **zoom máximo** para o provedor de blocos específico.

A visualização inicial é ignorada se a visualização padrão estiver definida (consulte abaixo).

Exemplos:

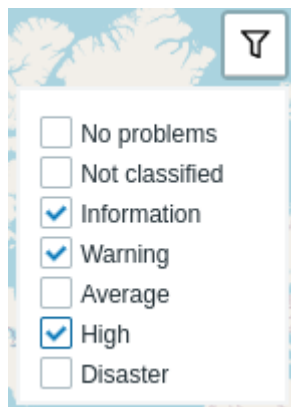
=> 40.6892494,-74.0466891,14

=> 40.6892494,-122.0466891

Visão inicial

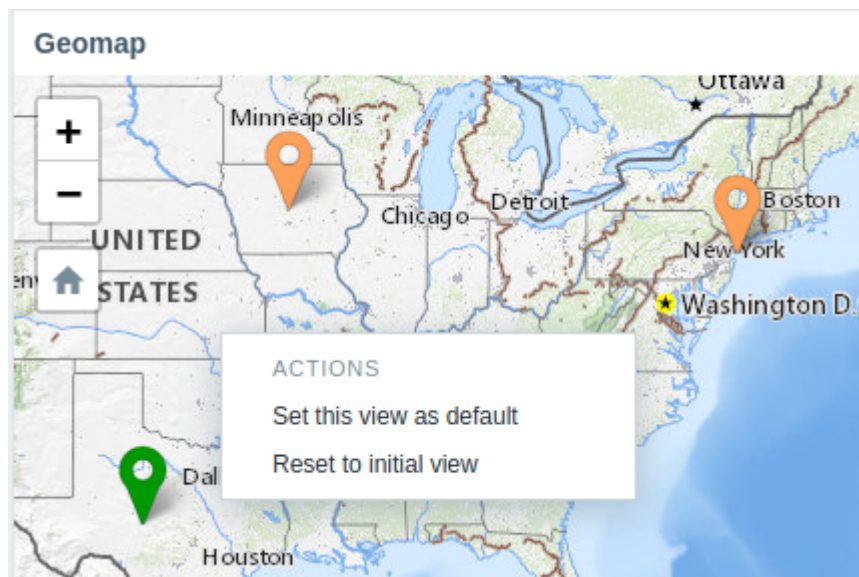
Os marcadores de host exibidos no mapa têm a cor do host mais problema sério e cor verde se um host não tiver problemas. Clicando em um marcador de host permite visualizar o nome visível do host e o número de problemas não resolvidos agrupados por gravidade. Clicando no nome visível vai abrir **host menu**.

Os hosts exibidos no mapa podem ser filtrados por gravidade do problema. Pressione o ícone de filtro no canto superior direito do widget e marque o gravidades.



É possível ampliar e reduzir o mapa usando o sinal de mais e menos botões no canto superior esquerdo do widget ou usando a rolagem do mouse volante ou touchpad. Para definir a visualização atual como padrão, clique com o botão direito do mouse em qualquer lugar no mapa e selecione Definir esta visualização como padrão. Esta configuração substituirá o parâmetro de widget Visualização inicial para o usuário atual. Para desfazer esta ação, clique com o botão direito em qualquer lugar do mapa novamente e selecione Redefinir a visualização inicial.

Quando Visualização inicial ou Visualização padrão estiver definida, você poderá retornar a esta visualizar a qualquer momento pressionando o ícone inicial à esquerda.



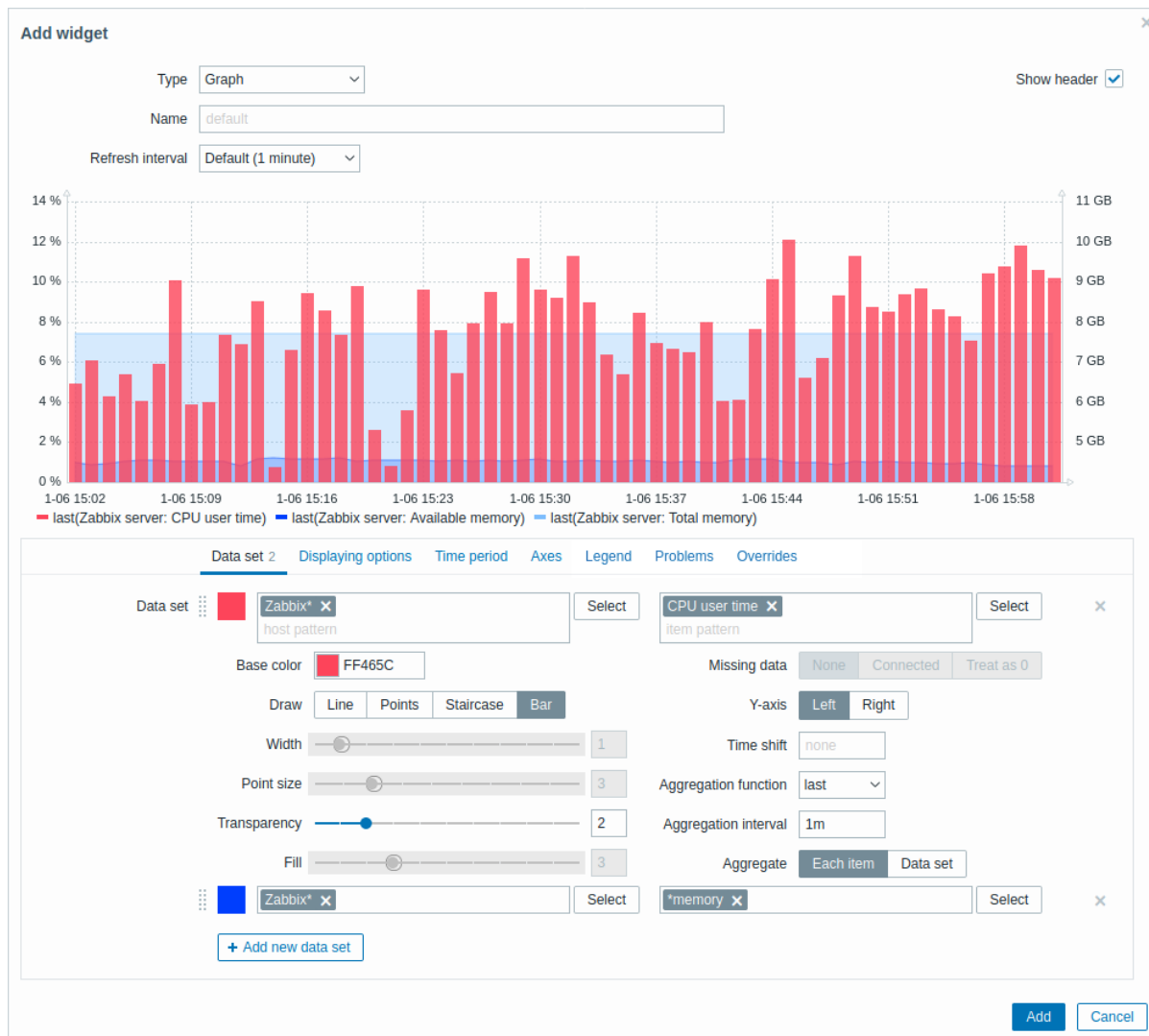
8 Gráfico

Visão geral

O widget de gráfico fornece uma maneira moderna e versátil de visualizar dados coletados pelo Zabbix usando uma técnica de desenho de imagens vetoriais. Este gráfico widget é suportado desde o Zabbix 4.0. Observe que o widget de gráfico suportado antes do Zabbix 4.0 ainda pode ser usado como **Graph (clássico)**.

Configuração

Para configurar, selecione Graph como tipo:



A guia **Conjunto de dados** permite adicionar conjuntos de dados e definir sua visualização:

Conjunto de dados




Selecione hosts e itens para exibir no gráfico. Alternativamente, você pode inserir padrões de host e item. Padrões curinga podem ser usados (por exemplo, * retornará resultados que correspondem a zero ou mais caracteres). Para especificar um padrão curinga, basta inserir a string manualmente e pressionar Enter. Enquanto você digita, observe como todos os hosts correspondentes são exibidos no menu suspenso. Até 50 itens podem ser exibidos no gráfico.

Os campos de padrão de host e padrão de item são obrigatórios. O símbolo curinga é sempre interpretado, portanto não é possível adicionar, por exemplo, um item chamado "item*" individualmente, se houver outros itens correspondentes (por exemplo, item2, item3).

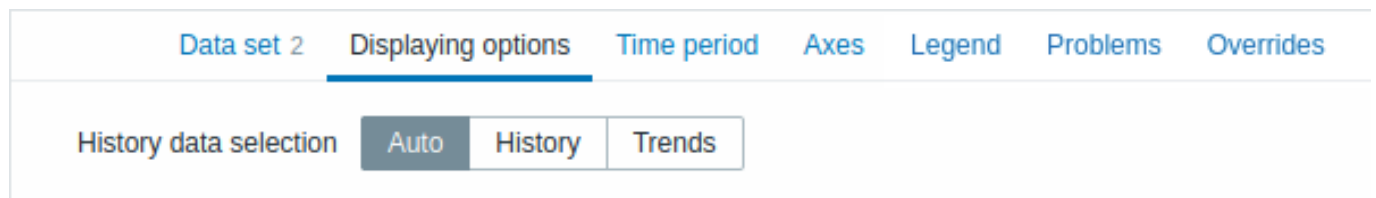
Cor base	Ajuste a cor base, a partir do seletor de cores ou manualmente. A cor base é usada para calcular cores diferentes para cada item do conjunto de dados. O campo de entrada de cor base é obrigatório.
Draw	Escolha o tipo de desenho da métrica. Os tipos de desenho possíveis são Linha (definido por padrão), Pontos, Escada e Barra. Observe que, se houver apenas um ponto de dados no gráfico de linha/escada, ele será desenhado como um ponto, independentemente de o tipo de sorteio. O tamanho do ponto é calculado a partir da largura da linha, mas não pode ser menor que 3 pixels, mesmo que a largura da linha seja menor.
Width	Defina a largura da linha. Esta opção está disponível quando o tipo de desenho Linha ou Escada é selecionado.
Tamanho do ponto	Defina o tamanho do ponto. Esta opção está disponível quando o tipo de desenho Pontos é selecionado.
Transparência Preenchimento	Defina o nível de transparência. Defina o nível de preenchimento. Esta opção está disponível quando o tipo de desenho Linha ou Escada é selecionado.
Dados ausentes	Selecione a opção para exibir dados ausentes: Nenhum - a lacuna é deixada em branco Conectado - dois valores de borda estão conectados Tratar como 0 - os dados ausentes são exibidos como valores 0 Não aplicável para os tipos de desenho Pontos e Barra.
Eixo Y	Selecione o lado do gráfico onde o eixo Y será exibido.
Deslocamento de horário	Especifique o deslocamento de horário, se necessário. Você pode usar sufixos de tempo neste campo. Valores negativos são permitidos.

Função de agregação	<p>Especifique qual função de agregação usar:</p> <p>min - exibe o menor valor</p> <p>max - exibe o maior valor</p> <p>avg - exibir o valor médio</p> <p>sum - exibir a soma dos valores</p> <p>count - exibir a contagem de valores</p> <p>first - exibir o primeiro valor</p> <p>** last** - exibe o último valor</p> <p>none - exibe todos os valores (sem agregação)</p> <p>A agregação permite exibir um valor agregado para o intervalo escolhido (5 minutos, uma hora, um dia), em vez disso de todos os valores. Veja também: Agregação em gráficos. Esta opção é suportada desde o Zabbix 4.4.</p>
Intervalo de agregação	<p>Especifique o intervalo para agregar valores. Você pode usar sufixos de tempo neste campo. Um valor numérico sem sufixo será considerado como segundos.</p> <p>Esta opção é suportada desde o Zabbix 4.4.</p>
Agregar	<p>Especifique se deseja agregar:</p> <p>Cada item - cada item no conjunto de dados será agregado e exibido separadamente.</p> <p>Conjunto de dados - todos os itens do conjunto de dados serão agregados e exibido como um valor.</p> <p>Esta opção é suportada desde o Zabbix 4.4.</p>

Os conjuntos de dados existentes são exibidos em uma lista. Você pode:

-  - clique neste botão para adicionar um novo conjunto de dados
-  - clique no ícone de cor para expandir/recolher os detalhes do conjunto de dados
-  - clique no ícone mover e arraste um conjunto de dados para um novo local na Lista

A guia **Exibindo opções** permite definir a seleção de dados do histórico:



Seleção de dados do histórico

Defina a origem dos dados do gráfico:

Auto - os dados são originados de acordo com o gráfico clássico **algoritmo** (padrão)

Histórico - dados do histórico

Tendências - dados das tendências

A guia **Período de tempo** permite definir um período de tempo personalizado:

Data set 2
Displaying options
Time period ●
Axes
Legend
Problems
Overrides

Set custom time period ☒

From

To

Definir período de tempo personalizado	Marque esta caixa de seleção para definir o período de tempo personalizado para o gráfico (desmarcado por padrão).
De	Defina a hora de início do período de tempo personalizado para o gráfico.
Para	Defina a hora de término do período de tempo personalizado para o gráfico.

A guia **Eixos** permite personalizar como os eixos são exibidos:

Data set 2
Displaying options
Time period
Axes ●
Legend
Problems
Overrides

Left Y ☒ Show

Right Y ☒ Show

X-Axis ☒ Show

Min

Max

Units

Min

Max

Units

Left Y	Marque esta caixa de seleção para tornar o eixo Y esquerdo visível. A caixa de seleção pode ser desativada se desmarcada em Conjunto de dados ou na guia Substituições.
Right Y	Marque esta caixa de seleção para tornar o eixo Y direito visível. A caixa de seleção pode ser desativada se desmarcada em Conjunto de dados ou na guia Substituições.
X-Axis	Desmarque esta caixa de seleção para ocultar o eixo X (marcado por padrão).
Min	Defina o valor mínimo do eixo correspondente. O valor mínimo do intervalo visível do eixo Y é especificado.
Max	Defina o valor máximo do eixo correspondente. O valor máximo do intervalo visível do eixo Y é especificado.
Unidades	Escolha a unidade para os valores do eixo do gráfico na lista suspensa. Se a opção Auto for escolhida, os valores dos eixos são exibidos usando as unidades do primeiro item do eixo correspondente. A opção Static permite atribuir o nome personalizado do eixo correspondente. Se a opção Static for escolhida e o campo de entrada value for deixado em branco, o nome do eixo correspondente consistirá apenas de um valor numérico.

A guia **Legenda** permite personalizar a legenda do gráfico:

Data set 2
Displaying options
Time period
Axes
Legend
Problems
Overrides

Show legend ☒

Number of rows

Mostrar legenda

Desmarque esta caixa de seleção para ocultar a legenda no gráfico (marcada por padrão).

Número de linhas

Defina o número de linhas a serem exibidas no gráfico.

A guia **Problemas** permite personalizar a exibição do problema:

Data set 2
Displaying options
Time period
Axes
Legend
Problems
Overrides

Show problems ☒

Selected items only ☒

Problem hosts

Severity
☐ Not classified
☐ Warning
☒ High
☐ Information
☐ Average
☒ Disaster

Problem

Tags
And/Or Or

Add

Mostrar problemas

Marque esta caixa de seleção para habilitar a exibição do problema no gráfico (desmarcado, ou seja, desabilitado por padrão).

Apenas itens selecionados

Marque esta caixa de seleção para incluir problemas apenas para os itens selecionados a serem exibidos no gráfico. Selecione os hosts com problemas a serem exibidos no gráfico. Padrões curinga podem ser usados (por exemplo, * retornará resultados que correspondem a zero ou mais caracteres). Para especificar um padrão curinga, basta inserir a string manualmente e pressionar Enter. Enquanto você digita, observe como todos os hosts correspondentes são exibidos na lista suspensa.

Gravidade

Marque as gravidades do problema a serem exibidas no gráfico.

Problema

Especifique o nome do problema a ser exibido no gráfico.

Especifique as tags de problema para limitar o número de problemas exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.

Há vários operadores disponíveis para cada condição:

Exists - inclua os nomes de tags especificados

Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)

Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)

Não existe - exclui o especificado nomes de tag

Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)

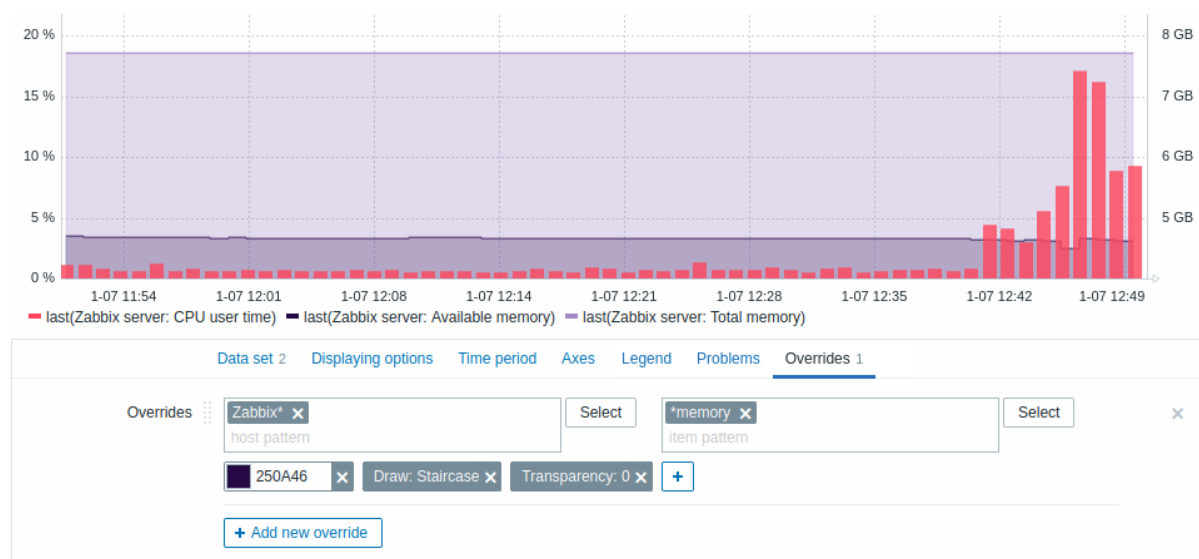
Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)

Existem dois tipos de cálculo para condições:

E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição

Ou - suficiente se uma condição for atendida

A guia **Substituições** permite adicionar substituições personalizadas para conjuntos de dados:

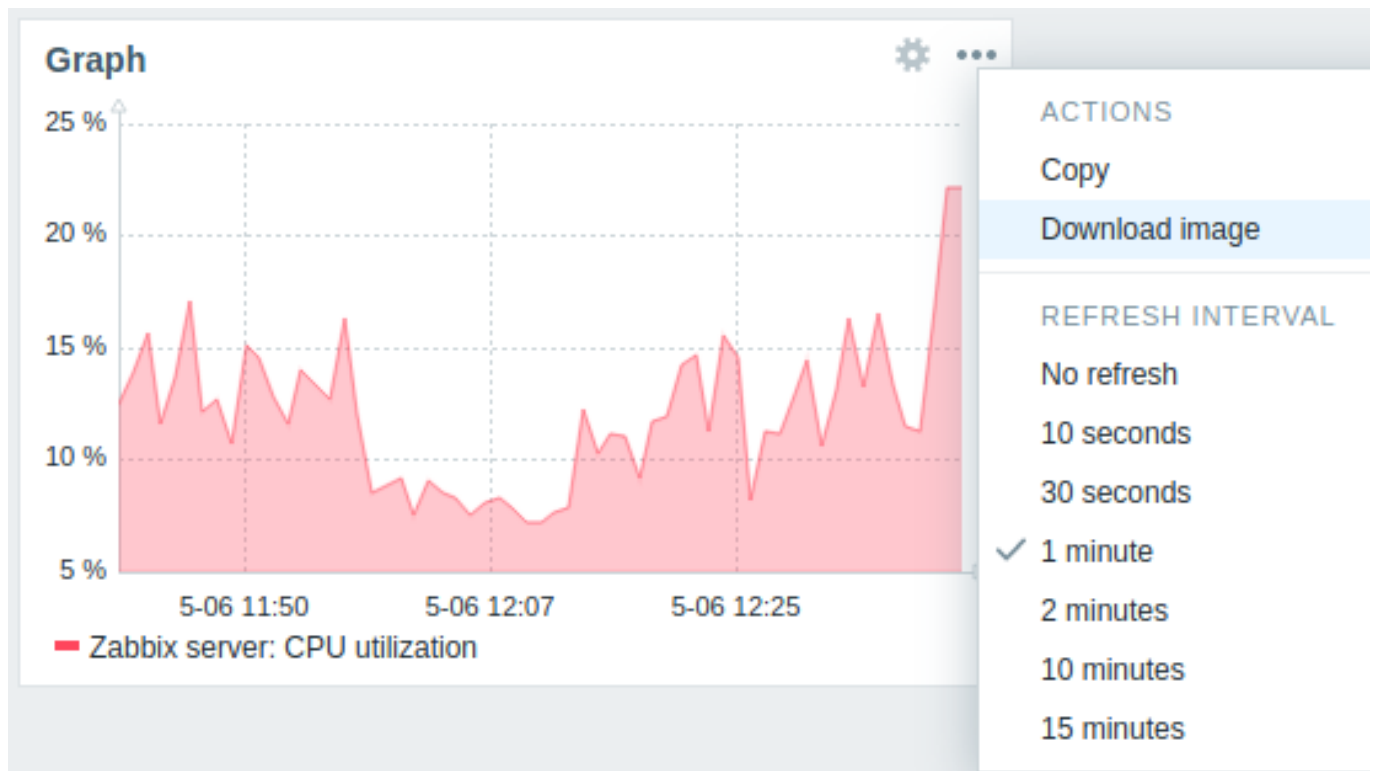


As substituições são úteis quando vários itens são selecionados para um conjunto de dados usando o curinga * e você deseja alterar como os itens são exibido por padrão (por exemplo, cor base padrão ou qualquer outra propriedade).

As substituições existentes (se houver) são exibidas em uma lista. Para adicionar um novo sobrepor:

- Clique no **+ Add new override** botão
- Selecione hosts e itens para a substituição. Alternativamente, você pode insira os padrões de host e item. Padrões curinga podem ser usados (para exemplo, * retornará resultados que correspondem a zero ou mais personagens). Para especificar um padrão curinga, basta inserir a string manualmente e pressione Enter. Enquanto você digita, observe como todos hosts correspondentes são exibidos no menu suspenso. O símbolo curinga é sempre interpretado, portanto não é possível acrescentar, por exemplo, um item chamado "item*" individualmente se houver outros itens correspondentes (por exemplo, item2, item3). Padrão de host e padrão de item Os campos são de preenchimento obrigatório.
- Clique em **+**, para selecionar parâmetros de substituição. Pelo menos um parâmetro de substituição deve ser selecionado. Para descrições de parâmetros, consulte o Conjunto de dados guia acima.

As informações exibidas pelo widget de gráfico podem ser baixadas como um .png imagem usando o **widget menu**:



Uma captura de tela do widget será salva na pasta Downloads.

9 Gráfico (clássico)

Visão geral

No widget de gráfico clássico, você pode exibir um único gráfico personalizado ou gráfico simples.

Configuração

Para configurar, selecione Gráfico (clássico) como tipo:

Add widget

Type

Graph (classic) ▾

Show header ☒

Name

System load

Refresh interval

Default (1 minute) ▾

Source

Graph Simple graph

* Graph

Zabbix server: System load ✕

Select

Show legend ☒

Dynamic item ☐

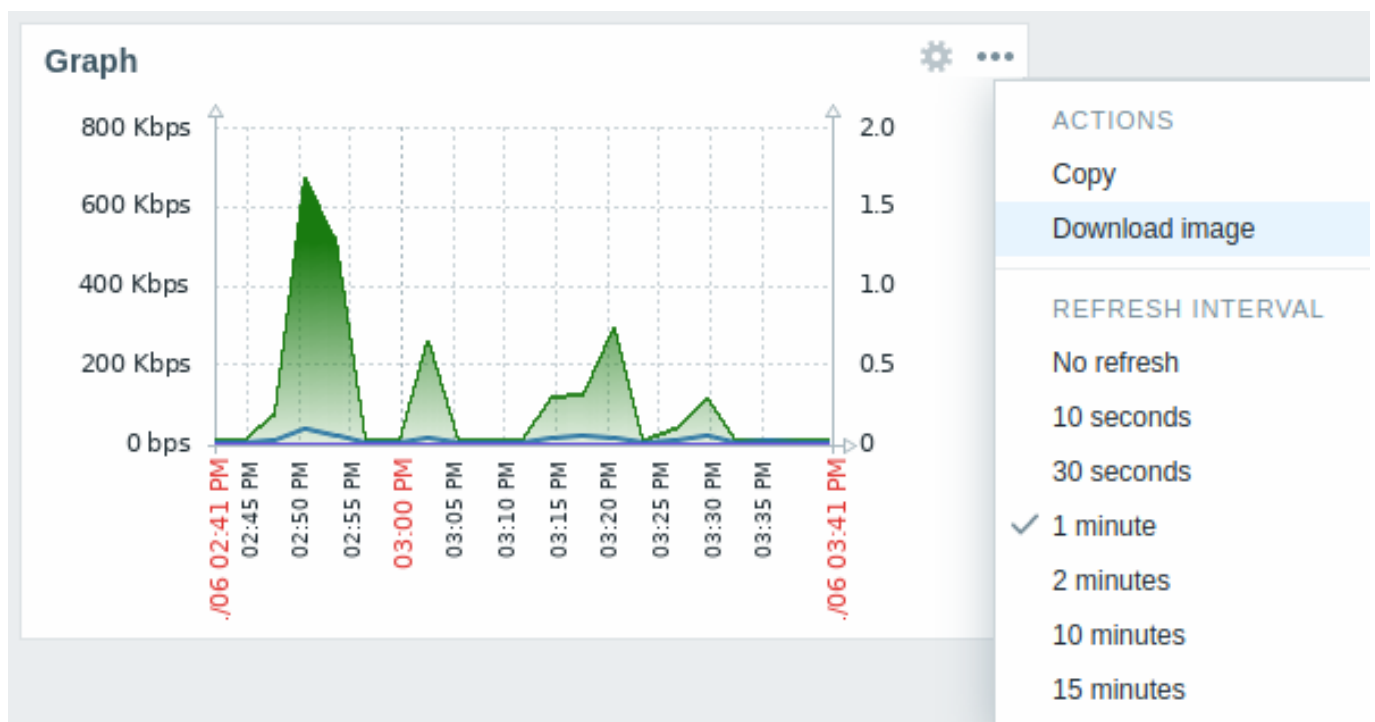
Add

Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Fonte	<p>Selecione o tipo de gráfico:</p> <p>Gráfico - gráfico personalizado</p> <p>Gráfico simples - gráfico simples</p>
Gráfico	<p>Selecione o gráfico personalizado a ser exibido.</p> <p>Esta opção estará disponível se 'Gráfico' for selecionado como Fonte.</p>
Item	<p>Selecione o item para exibir em um gráfico simples.</p> <p>Esta opção está disponível se 'Gráfico simples' estiver selecionado como Fonte.</p>
Mostrar legenda	<p>Desmarque esta caixa de seleção para ocultar a legenda no gráfico (marcada por padrão).</p>
Item dinâmico	<p>Configurar gráfico para exibir dados diferentes dependendo do host selecionado.</p>

As informações exibidas pelo widget de gráfico clássico podem ser baixadas como imagem .png usando o **widget menu**:



Uma captura de tela do widget será salva na pasta Downloads.

10 Protótipo de gráfico

Visão geral

No widget de protótipo de gráfico, você pode exibir uma grade de gráficos criados de um protótipo de gráfico ou de um protótipo de item por descoberta.

Configuração

Para configurar, selecione Protótipo de gráfico como tipo de widget:

Add widget

Type

Graph prototype

Show header

☒

Name

Graph prototype

Refresh interval

Default (1 minute)

Source

Graph prototype

Simple graph prototype

* Graph prototype

Zabbix server: Interface {#IFNAME}: Network traffic

Select

Show legend

☒

Dynamic item

☐

* Columns

2

* Rows

1

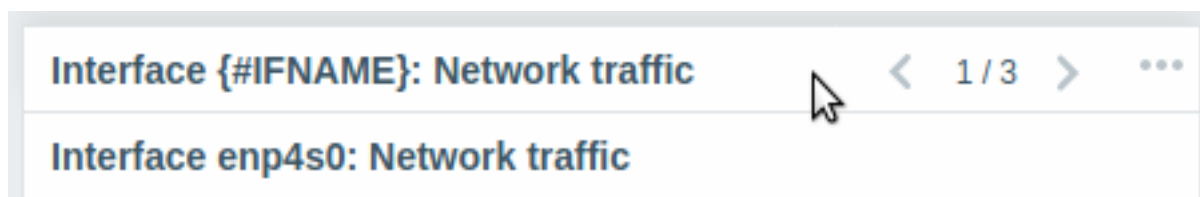
Add

Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Fonte	Selecione a fonte: um Protótipo de gráfico ou um Protótipo de gráfico simples .
Protótipo de gráfico	Selecione um protótipo de gráfico para exibir os gráficos descobertos do protótipo de gráfico. Esta opção está disponível se 'Protótipo de gráfico' for selecionado como Origem.
Protótipo de item	Selecione um protótipo de item para exibir gráficos simples com base em itens descobertos de um protótipo de item. Esta opção está disponível se 'Protótipo de gráfico simples' for selecionado como Origem.
Mostrar legenda	Marque esta caixa de seleção para mostrar a legenda nos gráficos (marcados por padrão).
Dynamic item	Defina gráficos para exibir dados diferentes dependendo do host selecionado.
Colunas	Insira o número de colunas de gráficos a serem exibidos em um widget de protótipo de gráfico.
Linhas	Digite o número de linhas de gráficos a serem exibidos em um widget de protótipo de gráfico.

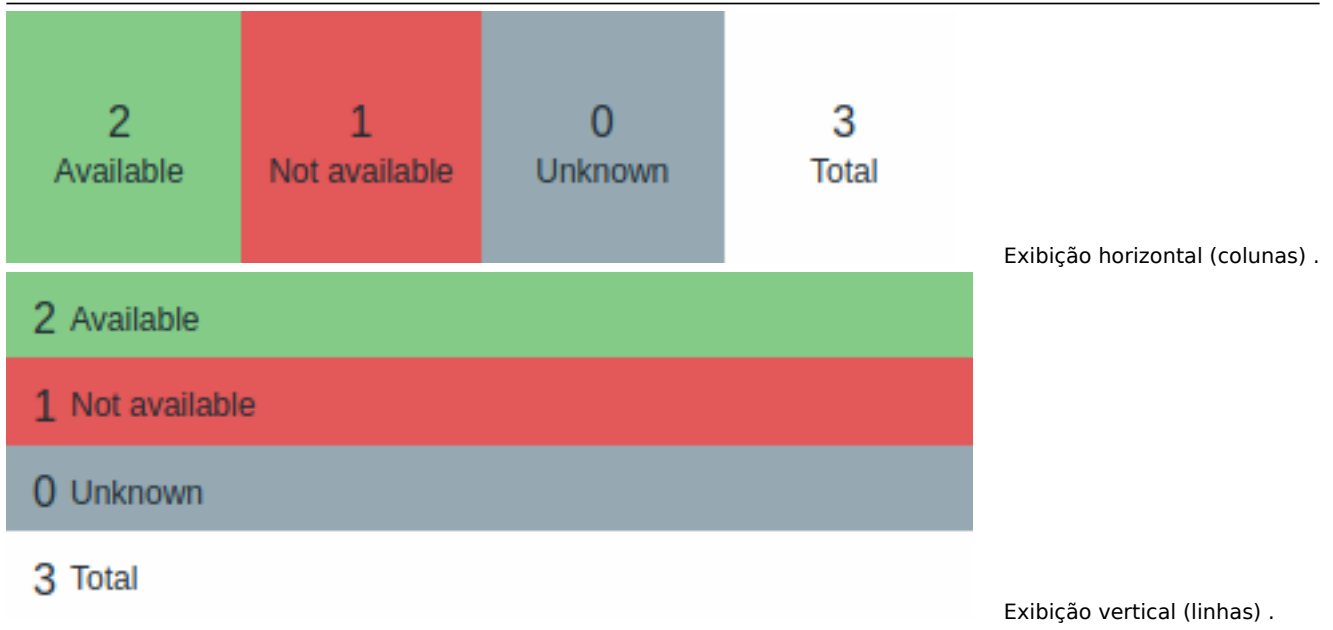
Enquanto as configurações de Colunas e Linhas permitem ajustar mais de uma gráfico no widget, ainda pode haver mais gráficos descobertos do que existem colunas/linhas no widget. Neste caso, a paginação torna-se disponível no widget e um cabeçalho deslizante permite alternar entre páginas usando as setas esquerda e direita.



11 Disponibilidade do host

Visão geral

No widget de disponibilidade do host, estatísticas de alto nível sobre o host disponibilidade são exibidos em quatro colunas/linhas coloridas.



A disponibilidade do host em cada coluna/linha é contada da seguinte forma:

- Disponível - hosts com todas as interfaces disponíveis
- Não disponível - hosts com pelo menos uma interface indisponível
- Desconhecido - hosts com pelo menos uma interface desconhecida (nenhuma indisponível)
- Total - total de todos os hosts

Configuração

Para configurar, selecione Disponibilidade do host como tipo:

×

Add widget

Type

Host availability

▼

Show header

☒

Name

Host availability

Refresh interval

Default (15 minutes)

▼

Host groups

Zabbix servers

×

type here to search

Select

Interface type

☒ Zabbix agent
 ☐ SNMP
 ☐ JMX
 ☐ IPMI

Layout

Horizontal

Vertical

Show hosts in maintenance

☐

Add

Cancel

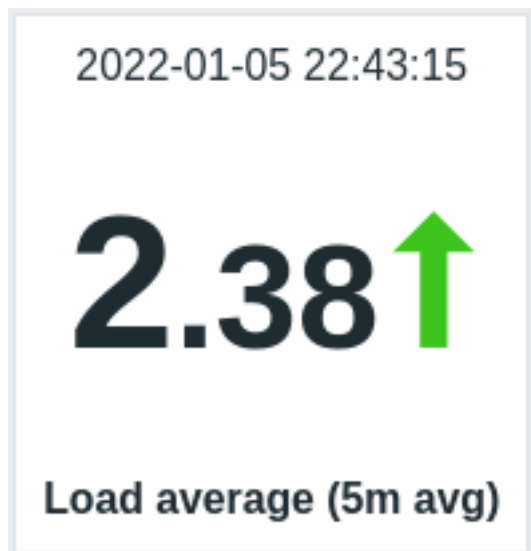
Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Grupos de hosts	Selecione o(s) grupo(s) de hosts. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes. Role para baixo para selecionar. Clique em 'x' para remover o selecionado.
Tipo de interface	Selecione para quais interfaces de host você deseja ver os dados de disponibilidade. A disponibilidade de todas as interfaces é exibida por padrão se nada for selecionado.
Layout	Selecione exibição horizontal (colunas) ou exibição vertical (linhas).
Mostrar hosts em manutenção	Incluir hosts que estão em manutenção nas estatísticas.

12 Valor do item

Visão geral

Este widget é útil para exibir o valor de um único item com destaque.



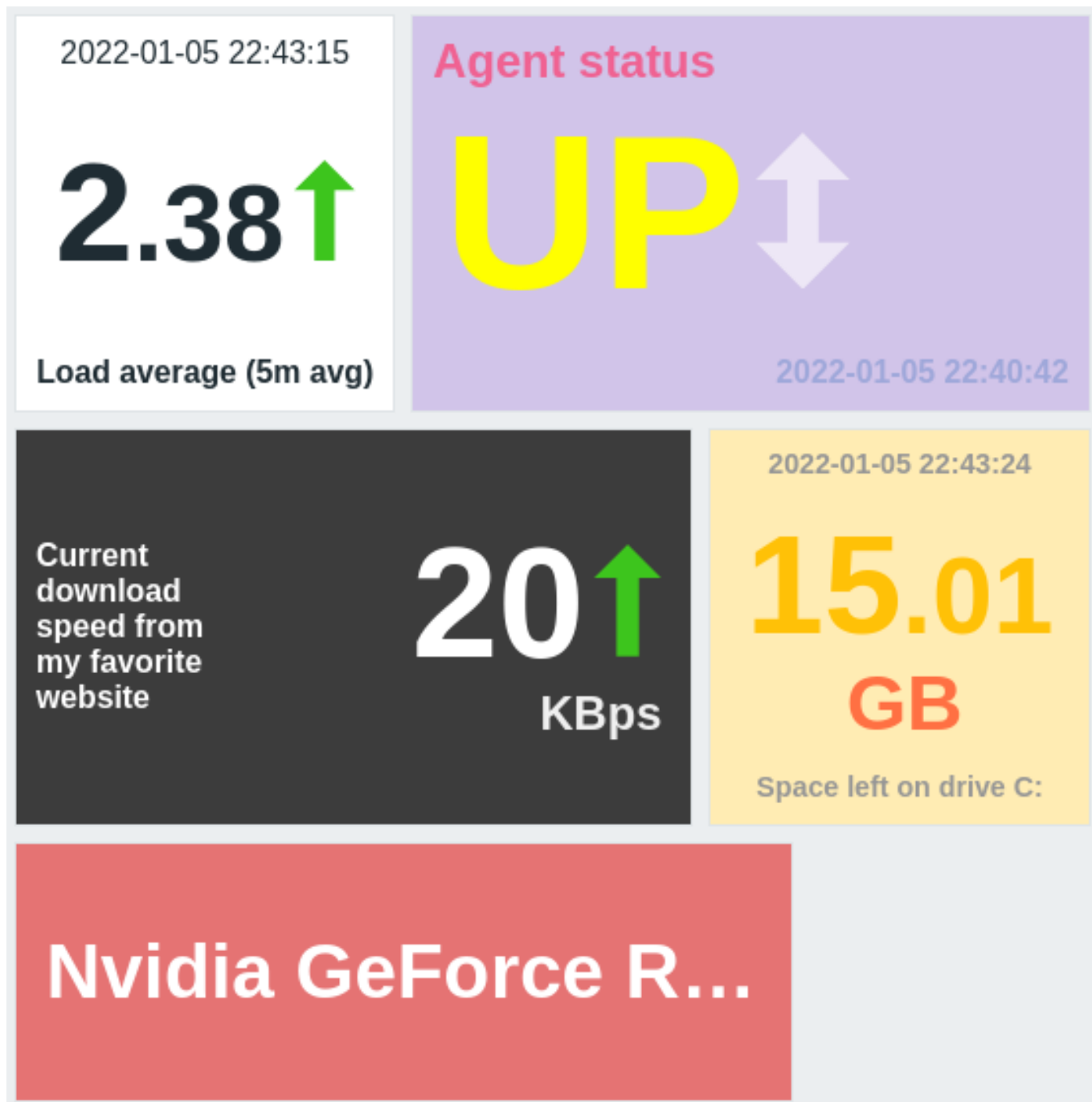
Além do valor em si, elementos adicionais podem ser exibidos, se desejado:

- tempo da métrica
- descrição do item
- indicador de mudança para o valor
- unidade de itens

O widget pode exibir valores numéricos e de string. Os valores de string são exibidos em uma única linha e truncados, se necessário. "Nenhum dado" é exibido, se não houver valor para o item.

Clicar no valor leva a um gráfico ad-hoc para itens numéricos ou dados mais recentes para itens de string.

O widget e todos os elementos nele podem ser ajustados visualmente usando as opções de **configuração avançada**, permitindo criar uma grande variedade de estilos visuais:



Configuração

Para configurar, selecione Valor do item como o tipo de widget:

×

Edit widget

Type

Item value

▼

Show header

☒

Name

Item value

Refresh interval

Default (1 minute)

▼

* Item

New host: CPU load average

×

Select

* Show

☒ Description

☒ Value

☒ Time

☒ Change indicator

Advanced configuration

☐

Dynamic item

☐

Apply

Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Item	Selecione o item.
Mostrar	Marque a caixa de seleção para exibir o respectivo elemento (descrição, valor, hora, indicador de alteração). Desmarque para ocultar.
Configuração avançada	Pelo menos um elemento deve ser selecionado. Marque a caixa de seleção para exibir as opções de configuração avançada .
Dynamic item	Marque a caixa de seleção para exibir um valor diferente dependendo do host selecionado.

Configuração avançada

As opções de configuração avançada ficam disponíveis se o Advanced configuração está marcada (veja a captura de tela) e apenas para os elementos selecionados no campo Show (veja acima).

Além disso, a configuração avançada permite alterar o plano de fundo cor para todo o widget.

Advanced configuration



* Description ?

{ITEM.NAME}

Horizontal position

Left

Center

Right

Size

15

%

Vertical position

Top

Middle

Bottom

Bold

☐

Color

D

Value

Decimal places

2

Size

35

%

Horizontal position

Left

Center

Right

Size

45

%

Vertical position

Top

Middle

Bottom

Bold

☒

Color

D



Units

Position

After value



Size

35

%

Bold

☒

Color

D

Time

Horizontal position

Left

Center

Right

Size

15

%

Vertical position

Top

Middle

Bottom

Bold

☐

Color

D

Change indicator



D



D



D

Background color

D

Descrição

Digite a descrição do item. Essa descrição pode substituir o nome do item padrão. Descrições de várias linhas são suportadas. É possível uma combinação de texto e macros compatíveis.

{HOST.*}, {ITEM.*}, {INVENTORY.*} e macros de usuário são compatíveis.

Posição horizontal

Selecione a posição horizontal da descrição do item - esquerda, direita ou centro.

Posição vertical

Selecione a posição vertical da descrição do item - superior, inferior ou central.

Tamanho

Digite a altura do tamanho da fonte para a descrição do item (em porcentagem relativa à altura total do widget).

Negrito

Marque a caixa de seleção para exibir a descrição do item em negrito.

Cor

Selecione a cor da descrição do item no seletor de cores. D significa a cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores.

Valor	
Casas decimais	Selecione quantas casas decimais serão exibidas com o valor. Este valor afetará apenas os itens flutuantes.
Tamanho	Insira a altura do tamanho da fonte para as casas decimais (em porcentagem em relação à altura total do widget).
Posição horizontal	Selecione a posição horizontal do valor do item - esquerda, direita ou centro.
Posição vertical	Selecione a posição vertical do valor do item - superior, inferior ou central.
Tamanho	Insira a altura do tamanho da fonte para o valor do item (em porcentagem em relação à altura total do widget). Observe que o tamanho do valor do item é priorizado; outros elementos têm que ceder espaço para o valor. No entanto, com o indicador de alteração, se o valor for muito grande, ele será truncado para mostrar o indicador de alteração.
Negrito	Marque a caixa de seleção para exibir o valor do item em negrito.
Cor	Selecione a cor do valor do item no seletor de cores. D significa a cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores.
Unidades	Marque a caixa de seleção para exibir as unidades com o valor do item. Se você inserir um nome de unidade, ele substituirá a unidade da configuração do item.
Posição	Selecione a posição da unidade do item - acima, abaixo, antes ou depois do valor.
Tamanho	Digite a altura do tamanho da fonte para a unidade do item (em porcentagem em relação à altura total do widget).
Negrito	Marque a caixa de seleção para exibir a unidade do item em negrito.
Cor	Selecione a cor da unidade do item no seletor de cores. D significa cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores.
Hora	
Posição horizontal	Hora é o valor do relógio do histórico do item. Selecione a posição horizontal do tempo - esquerda, direita ou centro.
Posição vertical	Selecione a posição vertical do tempo - superior, inferior ou central.
Tamanho	Digite a altura do tamanho da fonte para o tempo (em porcentagem em relação à altura total do widget).
Negrito	Marque a caixa de seleção para exibir a hora em negrito.
Cor	Selecione a cor do tempo no seletor de cores. D significa a cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores.
Indicador de mudança	
	Selecione a cor dos indicadores de mudança no seletor de cores. Os indicadores de mudança são os seguintes: ↑ - o valor do item está alto (para itens numéricos) ↓ - o valor do item está baixo (para itens numéricos) **↑** - o valor do item foi alterado (para itens de string e itens com mapeamento de valor) D significa cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores. O tamanho vertical do indicador de alteração é igual ao tamanho do valor (parte inteira do valor para itens numéricos). Observe que os indicadores para cima e para baixo não são mostrados com apenas um valor.

Cor de fundo	<p>Selecione a cor de fundo para todo o widget no seletor de cores.</p> <p>D significa cor padrão (depende do tema do frontend). Para retornar ao valor padrão, clique no botão Usar padrão no seletor de cores.</p>
---------------------	--

Observe que vários elementos não podem ocupar o mesmo espaço; se forem colocados no mesmo espaço, será exibida uma mensagem de erro.

13 Mapa

Visão Geral

No widget de mapa, você pode exibir:

- um único mapa de rede configurado;
- um dos mapas de rede configurados na **map navigation tree** (ao clicar no nome do mapa na árvore).

Configuração

Para configurar, selecione o Map como tipo:

Além dos parâmetros que são comuns para todos os widgets **common**, você pode definir as seguintes opções:

Tipo de fonte	<p>Selecione para exibir:</p> <p>Mapa - mapa de rede;</p> <p>Árvore de navegação do mapa - um dos mapas na árvore de navegação do mapa selecionado.</p>
Mapa	<p>Selecione o mapa a ser exibido.</p> <p>Este campo é de auto-completar, então se você começar a digitar o nome de um mapa oferecerá um menu suspenso de mapas correspondentes.</p> <p>Esta opção está disponível se 'Mapa' for selecionado como Tipo de fonte.</p>
Filtro	<p>Selecione a árvore de navegação do mapa para exibir os mapas.</p> <p>Esta opção está disponível se 'Árvore de navegação do mapa' for selecionada como Tipo de fonte.</p>

14 Árvore de navegação do mapa

Visão geral

Este widget permite construir uma hierarquia de mapas existentes enquanto também exibindo estatísticas de problemas com cada mapa incluído e grupo de mapas.

Torna-se ainda mais poderoso se você vincular o widget Mapa ao árvore de navegação. Neste caso, clicar em um nome de mapa na navegação árvore exibe o mapa completo no widget Mapa.



As estatísticas com o mapa de nível superior na hierarquia exibem uma soma de problemas de todos os sub-mapas e seus próprios problemas.

Configuração

Para configurar o widget da árvore de navegação, selecione Árvore de navegação do mapa como tipo:

The 'Add widget' dialog box is shown. It has a close button (X) in the top right corner. The 'Type' dropdown menu is set to 'Map navigation tree'. The 'Name' text field contains 'Map tree'. The 'Refresh interval' dropdown menu is set to 'Default (15 minutes)'. The 'Show unavailable maps' checkbox is unchecked. The 'Show header' checkbox is checked. At the bottom right, there are 'Add' and 'Cancel' buttons.

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Mostrar mapas indisponíveis

Marque esta caixa de seleção para exibir mapas para os quais o usuário não tem permissão de leitura. Os mapas indisponíveis na árvore de navegação serão exibidos com um ícone acinzentado. Observe que, se esta caixa de seleção estiver marcada, os submapas disponíveis serão exibidos mesmo se o mapa de nível pai estiver indisponível. Se desmarcados, os submapas disponíveis para um mapa pai indisponível não serão exibidos. A contagem de problemas é calculada com base nos mapas disponíveis e nos elementos do mapa disponíveis.

Navigation tree elements are displayed in a list. You can:

- drag an element (including its child elements) to a new place in the list;
- expand or collapse an element to display or hide its child elements;
- add a child element (with or without a linked map) to an element;
- add multiple child elements (with linked maps) to an element;
- edit an element;
- remove an element (including its child elements).

Map tree

root	<div><div>+</div><div></div></div>
<div><div></div><div>▼ All</div></div>	<div><div>+</div><div></div><div></div><div>×</div></div>
<div><div></div><div>Stockholm</div></div>	<div><div>+</div><div></div><div></div><div>×</div></div>
<div><div></div><div>London</div></div>	<div><div>+</div><div></div><div></div><div>×</div></div>
<div><div></div><div>► Munich</div></div>	<div><div>+</div><div></div><div></div><div>×</div></div>

Element configuration

To configure a navigation tree element, either add a new element or edit an existing element.

Edit tree element

* Name

London

Linked map

London network

×

Select

☐

Add submaps

Add

Cancel

The following navigation tree element configuration parameters are available:

Name	Enter the navigation tree element name.
Linked map	Select the map to link to the navigation tree element. This field is auto-complete so starting to type the name of a map will offer a dropdown of matching maps.
Add submaps	Mark this checkbox to add the submaps of the linked map as child elements to the navigation tree element.

15 Texto simples

Visão geral

No widget de texto simples, você pode exibir os dados mais recentes do item em formato simples texto.

Configuração

Para configurar, selecione Texto simples como tipo:

Add widget

Type

Plain text

Show header

☒

Name

Available memory

Refresh interval

Default (1 minute)

* Items

Zabbix server: Available memory

type here to search

Select

Items location

Left

Top

* Show lines

25

Show text as HTML

☐

Dynamic items

☐

Add

Cancel

Além dos parâmetros que são comuns para todos os widgets, você pode definir as seguintes opções específicas:

Itens	Selecione os itens.
Localização dos itens	Escolha a localização dos itens selecionados a serem exibidos no widget.
Mostrar linhas	Defina quantas linhas de dados mais recentes serão exibidas no widget.
Mostrar texto como HTML	Defina para exibir texto como HTML.
Dynamic item	Configurar para exibir dados diferentes dependendo do host selecionado.

16 Hosts com problema

Visão geral

No widget do host do problema, você pode exibir informações de alto nível sobre a disponibilidade do host.

Configuração

Para configurar, selecione Problema de hosts como tipo:

Add widget



Type
Problem hosts
Show header
☒

Name
Problem hosts

Refresh interval
Default (1 minute)

Host groups
Linux servers
Select

Exclude host groups
type here to search
Select

Hosts
type here to search
Select

Problem

Severity
☐ Not classified
☐ Warning
☐ High
☐ Information
☐ Average
☐ Disaster

Tags
And/Or Or
tag Contains value Remove
Add

Show suppressed problems
☐

Hide groups without problems
☐

Problem display
All Separated Unacknowledged only

Add Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Parâmetro	Descrição
Grupos de hosts	<p>Digite os grupos de hosts a serem exibidos no widget. Este campo é preenchido automaticamente, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes.</p> <p>A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.</p> <p>Os dados de host desses grupos de hosts serão exibidos no widget. Se nenhum grupo de hosts for inserido, todos os grupos de hosts serão exibidos.</p>
Excluir grupos de hosts	<p>Digite grupos de hosts para ocultar do widget. Este campo é preenchido automaticamente, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes.</p> <p>A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.</p> <p>Os dados de host desses grupos de hosts não serão exibido no widget. Por exemplo, os hosts 001, 002, 003 podem estar no Grupo A e os hosts 002, 003 no Grupo B também. Se selecionarmos mostrar o Grupo A e excluir o Grupo B ao mesmo tempo, somente os dados do host 001 serão exibidos no Painel.</p>

Parâmetro	Descrição
Hosts	<p>Digite os hosts a serem exibidos no widget. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um host oferecerá uma lista suspensa de hosts correspondentes. Se nenhum host for inserido, todos os hosts serão exibidos.</p>
Problema	<p>Você pode limitar o número de hosts problemáticos exibidos pelo nome do problema. Se você inserir uma string aqui, somente os hosts com problemas cujo nome contém a string inserida serão exibidos. As macros não são expandidas.</p>
Gravidade	<p>Marque as gravidades do problema a serem exibidas no widget.</p>
Tags	<p>Especifique as tags de problema para limitar o número de problemas exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.</p> <p>Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclua os nomes de tags especificados</p> <p>Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
Ou - suficiente se uma condição for atendida</p>
Mostrar problemas suprimidos	<p>Marque a caixa de seleção para exibir problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.</p>
Ocultar grupos sem problemas	<p>Marque a opção Ocultar grupos sem problemas para ocultar dados de grupos de hosts sem problemas no widget.</p>
Exibição do problema	<p>Exibir a contagem de problemas como:</p> <p>All - a contagem total de problemas será exibida</p> <p>Separated - a contagem de problemas não confirmada será exibida separada como um número da contagem total de problemas</p> <p>Apenas não confirmado - somente a contagem de problemas não confirmados será exibida.</p>

17 Problemas

Visão geral

Neste widget você pode exibir os problemas atuais. As informações neste widget é semelhante ao Monitoramento → Problemas.

Configuração

Para configurar, selecione Problemas como tipo:

Add widget

Type
Problems

Show header
☒

Name
Problems

Refresh interval
Default (1 minute)

Show
Recent problems
Problems
History

Host groups
type here to search
Select

Exclude host groups
type here to search
Select

Hosts
type here to search
Select

Problem

Severity
☐ Not classified
☐ Warning
☐ High
☐ Information
☐ Average
☐ Disaster

Tags
And/Or
Or

tag
Contains
value

Add

Show tags
None
1
2
3

Add
Cancel

Você pode limitar quantos problemas são exibidos no widget em várias maneiras - por status do problema, nome do problema, gravidade, grupo de hosts, host, tag de evento, status de confirmação, etc.

Parâmetro	Descrição
Mostrar	Filtrar por status do problema: Problemas recentes - problemas não resolvidos e resolvidos recentemente são exibidos (padrão) Problemas - problemas não resolvidos são exibidos **Histórico** - o histórico de todos os eventos é exibido
Grupos de hosts	Insira os grupos de hosts para exibir os problemas no widget. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados. Os problemas desses grupos de hosts serão exibidos em o widget. Se nenhum grupo de hosts for inserido, os problemas de todos os grupos de hosts serão exibidos.

Parâmetro	Descrição
Excluir grupos de hosts	<p>Insira grupos de hosts para ocultar problemas do widget. Este campo é preenchido automaticamente, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes.</p> <p>A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.</p> <p>Os problemas desses grupos de hosts não serão exibidos no widget. Por exemplo, os hosts 001, 002, 003 podem estar no Grupo A e os hosts 002, 003 no Grupo B também. Se selecionarmos mostrar o Grupo A e excluir o Grupo B ao mesmo tempo, apenas os problemas do host 001 serão exibidos no widget.</p>
Hosts	<p>Digite hosts para exibir problemas no widget. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um host oferecerá uma lista suspensa de hosts correspondentes. Se nenhum host for inserido, os problemas de todos os hosts serão exibidos.</p>
Problema	<p>Você pode limitar o número de problemas exibidos pelo nome. Se você inserir uma string aqui, somente os problemas cujo nome contém a string inserida serão exibidos. As macros não são expandidas.</p>
Gravidade	<p>Marque as gravidades do problema a serem exibidas no widget.</p>
Tags	<p>Especifique as tags de problema para limitar o número de problemas exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.</p> <p>Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclua os nomes de tags especificados</p> <p>Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
>Ou - suficiente se uma condição for atendida</p> <p>Quando filtradas, as tags especificadas aqui serão exibidas primeiro com o problema, a menos que substituídas pela Prioridade de exibição de tags (veja abaixo) ist.</p>
Mostrar tags	<p>Selecione o número de tags exibidas:</p> <p>Nenhuma - nenhuma coluna Tags em Monitoramento → Problemas</p> <p>1 - A coluna Tags contém uma tag
2 - A coluna Tags contém duas tags</p> <p>3 - A coluna Tags contém três tags</p> <p>Para ver todas as tags do problema, passe o mouse sobre o ícone de três pontos.</p>
Nome da tag	<p>Selecione o modo de exibição do nome da tag:</p> <p>Full - os nomes e valores das tags são exibidos por completo</p> <p>Encurtado - os nomes das tags são reduzidos para 3 símbolos; os valores das tags são exibidos na íntegra</p> <p>Nenhum - somente os valores das tags são exibidos; sem nomes</p>

Parâmetro	Descrição
Prioridade de exibição de tags	Digite a prioridade de exibição de tags para um problema, como uma lista de tags separadas por vírgula (por exemplo: Serviços,Aplicativos,Aplicativo). Devem ser usados apenas nomes de tags, sem valores. As tags desta lista sempre serão exibidas primeiro, substituindo a ordenação natural por alfabeto.
Mostrar dados operacionais	Selecione o modo para exibir dados operacionais : Nenhum - nenhum dado operacional é exibido ** Separadamente** - os dados operacionais são exibidos em uma coluna separada Com nome do problema - anexa dados operacionais ao nome do problema, usando parênteses para os dados operacionais
Mostrar problemas suprimidos	Marque a caixa de seleção para exibir problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.
Mostrar somente os não confirmados	Marque a caixa de seleção para exibir apenas os problemas não confirmados.
Ordenar entradas por	Ordenar entradas por: Hora (decrecente ou crescente) Gravidade (decrecente ou crescente) Nome do problema (decrecente ou crescente)) Host (descendente ou ascendente).
Mostrar linha do tempo	Marque a caixa de seleção para exibir uma linha do tempo visual.
Mostrar linhas	Especifique o número de linhas de problema a serem exibidas.

18 Problemas por severidade

Visão geral

Neste widget, você pode exibir problemas por severidade. Você pode limitar o que hosts e gatilhos são exibidos no widget e definem como o contagem de problemas é exibida.

Configuração

Para configurar, selecione Problemas por gravidade como tipo:

Add widget

Type
Problems by severity
Show header

Name
Problems by severity

Refresh interval
Default (1 minute)

Host groups
type here to search
Select

Exclude host groups
type here to search
Select

Hosts
type here to search
Select

Problem

Severity
☐ Not classified
☐ Warning
☐ High
☐ Information
☐ Average
☐ Disaster

Tags
And/Or Or
tag Contains value

Add

Show
Host groups Totals

Layout
Horizontal Vertical

Show additional data
None

Add Cancel

Além dos parâmetros que são **comuns** para todos os widgets, você pode definir as seguintes opções específicas:

Parâmetro	Descrição
Grupos de hosts	<p>Digite os grupos de hosts a serem exibidos no widget. Este campo é preenchido automaticamente, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes.</p> <p>A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.</p> <p>Os dados de host desses grupos de hosts serão exibidos no widget. Se nenhum grupo de hosts for inserido, todos os grupos de hosts serão exibidos.</p>

Parâmetro	Descrição
Excluir grupos de hosts	<p>Digite grupos de hosts para ocultar do widget. Este campo é preenchido automaticamente, portanto, começar a digitar o nome de um grupo oferecerá uma lista suspensa de grupos correspondentes.</p> <p>A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.</p> <p>Os dados de host desses grupos de hosts não serão exibido no widget. Por exemplo, os hosts 001, 002, 003 podem estar no Grupo A e os hosts 002, 003 no Grupo B também. Se selecionarmos mostrar o Grupo A e excluir o Grupo B ao mesmo tempo, somente os dados do host 001 serão exibidos no Painel.</p>
Hosts	<p>Digite os hosts a serem exibidos no widget. Este campo é de preenchimento automático, portanto, começar a digitar o nome de um host oferecerá uma lista suspensa de hosts correspondentes.</p> <p>Se nenhum host for inserido, todos os hosts serão exibidos.</p>
Problema	<p>Você pode limitar o número de hosts problemáticos exibidos pelo nome do problema. Se você inserir uma string aqui, somente os hosts com problemas cujo nome contém a string inserida serão exibidos. As macros não são expandidas.</p>
Gravidade	<p>Marque as gravidades do problema a serem exibidas no widget.</p>
Tags	<p>Especifique as tags de problema para limitar o número de problemas exibidos no widget. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.</p> <p>Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclua os nomes de tags especificados</p> <p>Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
Ou - suficiente se uma condição for atendida</p>
Mostrar	<p>Selecione a opção de exibição:</p> <p>Host groups - exibe problemas por grupo de hosts</p> <p>Totais - exibe um total de problemas para todos os grupos de hosts selecionados em blocos coloridos correspondentes ao problema gravidade.</p>
Layout	<p>Selecione a opção de layout:</p> <p>Horizontal - blocos coloridos de totais serão exibidos horizontalmente</p> <p>Vertical - blocos coloridos de totais serão exibidos verticalmente</p> <p>Este campo é disponível para edição se 'Totais' estiver selecionado como a opção Mostrar.</p>
Mostrar problemas suprimidos	<p>Marque a caixa de seleção para exibir problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.</p>
Ocultar grupos sem problemas	<p>Marque a opção Ocultar grupos sem problemas para ocultar dados de grupos de hosts sem problemas no widget.</p>
Mostrar dados operacionais	<p>Marque a caixa de seleção para exibir dados operacionais (consulte a descrição de Dados operacionais em Monitoring → Problems).</p>

Parâmetro	Descrição
Exibição do problema	Exibir a contagem de problemas como: All - a contagem total de problemas será exibida Separated - a contagem de problemas não confirmada será exibida separada como um número da contagem total de problemas Apenas não confirmado - somente a contagem de problemas não confirmados será exibida.
Mostrar linha do tempo	Marque a caixa de seleção para exibir uma linha do tempo visual.

19 Relatório SLA

Visão geral

Os relatórios de SLA também podem ser exibidos como um **widget de painel**. A funcionalidade é similar a seção Serviços -> Relatório SLA.

Configuração

Para configurar, selecione Relatório SLA com o tipo:

×

Type

SLA report

▼

Show header

☒

Name

SLA3

Refresh interval

Default (No refresh)

▼

* SLA

SLA:3

×

Select

Service

type here to search

Select

Show periods

15

From

YYYY-MM-DD

To

YYYY-MM-DD

Apply

Cancel

Além dos parâmetros que são **comuns** para todos widgets, você pode configurar as seguintes opções:

SLA	Seleciona o SLA para o relatório.
Serviço	Seleciona o serviço para o relatório.
Mostrar períodos	Seleciona quantos períodos serão mostrados no widget (20 por padrão, máximo de 100).
De	Seleciona a data inicial para o relatório. Datas relativas são suportadas: now, now/d, now/w-1w etc; suporte para datas modificadas: d, w, M, y.
Até	Seleciona a data final para o relatório. Datas relativas são suportadas: now, now/d, now/w-1w etc;suporte para datas modificadas: d, w, M, y.

20 Informações do sistema

This widget displays the same information as in Reports → **System information**, however, a single dashboard widget can only display either the system stats or the high availability nodes at a time (not both).

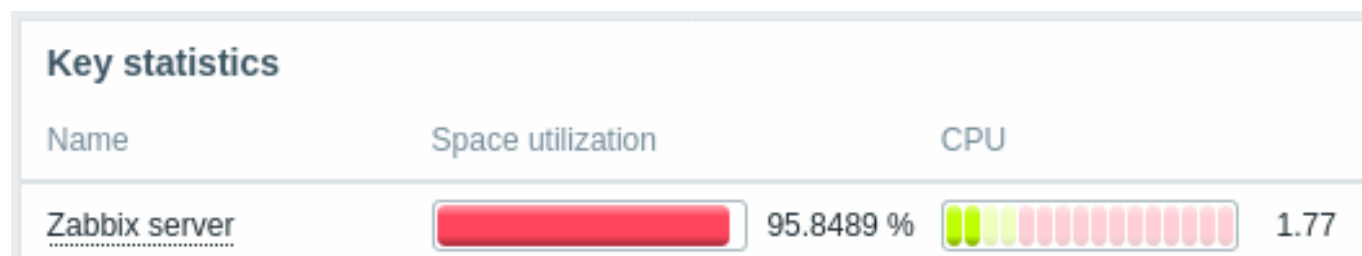
To configure, select System information as type:

All configuration parameters are **common** for all widgets.

Visão geral

Este widget fornece uma maneira de criar tabelas personalizadas para exibir a situação dos dados, permitindo exibir relatórios do tipo Top N e relatórios de barra de progresso úteis para o planejamento de capacidade.

O número máximo de hosts que pode ser exibido é 100.



Para configurar, seleccionar Top hosts como:

Add widget

Type

Top hosts

Show header

Name

default

Refresh interval

Default (1 minute)

Host groups

type here to search

Select

Hosts

type here to search

Select

Host tags

And/Or

Or

tag

Contains

value

Remove

Add

* Columns

Add

Order

Top N

Bottom N

* Order column

Add item column

* Host count

10

Add

Cancel

Além dos parâmetros que são comum para todos os widgets **common**, você pode configurar as opções específicas abaixo:

Host groups	Grupos host para exibir dados.
Hosts	Hosts para exibir dados.
Host tags	<p>Especifique as tags para limitar o número de hosts exibidos no widget. É possível incluir e excluir as tags específicas e os valores de tags. Diversas condições podem ser configuradas. O nome da tag correspondente é sempre sensível a maiúsculas e minúsculas.</p> <p>Existem diversos operadores disponíveis para cada condição:</p> <p>Existe - incluir os nomes de tags específicas</p> <p>Igual - incluir os nomes de tags específicas e valores de tag (maiúscula-minúscula).</p> <p>Contém - incluir os nomes de tags específicas quando os valores das tags contêm a string inserida (correspondência de subcadeia, não sensível a maiúsculas e minúsculas)</p> <p>Não existe - excluir os nomes de tags específicas</p> <p>Não é igual - excluir os nomes de tags específicas e valores de tag (maiúscula-minúscula)</p> <p>Não contém - excluir os nomes de tags específicas quando os valores das tags contêm a string inserida (correspondência de subcadeia, não sensível a maiúsculas e minúsculas)</p> <p>Existem dois tipos de cálculos para as condições:</p> <p>E/Ou - todas as condições deverão ser encontradas, as condições com o mesmo nome da tag serão agrupadas pela condição Ou</p> <p>Ou - suficiente se uma condição for encontrada</p>
Columns	<p>Adicione dados columns para exibição.</p> <p>A ordem da coluna determina sua exibição da direita para esquerda.</p> <p>As colunas podem ser reordenadas arrastando para cima e para baixo a alça antes do nome da coluna.</p>
Order	<p>Especifique a ordem das linhas:</p> <p>Parte superior N - em ordem decrescente pelo valor agregado da Ordenação da coluna</p> <p>Parte inferior N - em ordem crescente pelo valor agregado da Ordenação da coluna</p>
Order column	Especifique a coluna da lista de Colunas definidas a ser usada para a ordenação Top N ou Bottom N.

Host count	Contagem das linhas de host serão exibidas (1-100).
------------	---

Configuração da coluna

New column

Name

Data

Item value

* Item

type here to search

Select

Time shift

none

Aggregation function

none

Display

As is

Bar

Indicators

History data

Auto

History

Trends

Base color

Thresholds

Threshold

Add

Action

Add

Cancel

Parâmetros comuns da coluna:

Name	Nome da coluna
Data	Tipo de dado a ser exibido na coluna: Valor do item - valor do item especificado Host nome - o nome do host do item especificado na coluna Valor do item Texto - string de texto estático
Base color	Cor de fundo da coluna; preencher a cor se o dado Valor do item for exibido como barra/indicadores. Para o dado Valor do item, a cor padrão pode ser substituída por uma cor personalizada se o valor do item estiver acima do limite dos "Thresholds" especificados.

Parâmetros específicos para as colunas de valor do item:

Item	Selecione o item. A seleção de itens não numéricos é suportada desde a versão Zabbix 6.0.4.
Time shift	Especifique o tempo de intervalo se necessário. Você deve usar time suffixes neste campo. Valores negativos são permitidos.
Aggregation function	Especifique qual função de agregação utilizar: min - exibir o menor valor max - exibir o maior valor avg - exibir o valor médio sum - exibir a soma dos valores count - exibir a conta dos valores first - exibir o primeiro valor last - exibir o último valor none - exibir todos os valores (sem agregação) A Agregação permite exibir um valor agregado para o intervalo escolhido (5 minutos, 1 hora, 1 dia), em vez de todos os valores. Observe que apenas os itens numéricos podem ser exibidos nesta coluna se a configuração não estiver como "none".

Aggregation interval	Especifique o intervalo para valores de agregação. Você deve utilizar time suffixes neste campo. Um valor numérico sem um sufixo será estimado como segundos.
Display	Esse campo não será exibido se a Função de agregação estiver configurada como "none". Defina como o valor deve ser exibido: As is - como texto regular Bar - como sólido, barra preenchida com cor Indicators - como seguimento, barra preenchida com cor Observe que apenas os itens numéricos podem ser exibidos nessa coluna se a configuração estiver como "as is".
History	Obtenha dados a partir do histórico ou trends: Auto - seleção automática History - obtenha histórico de dados Trends - obtenha trend de dados Esta configuração se aplica apenas para dados numéricos. Dados não numéricos serão sempre obtidos no histórico.
Min	Valor mínimo para a barra/indicadores.
Max	Valor máximo para barra/indicadores.
Thresholds	Especifique valores de limite quando o fundo/preenchimento de cor deve ser alterada. A lista será classificada em ordem ascendente quando salva. Observe que apenas itens numéricos podem ser exibidos nesta coluna se limites forem utilizados.

Parâmetros específicos para colunas de texto:

Text	Digite a string a ser exibida. Deve conter host e inventário macros .
------	--

21 Visão geral de triggers

Overview

In the trigger overview widget, you can display the trigger states for a group of hosts.

- The trigger states are displayed as colored blocks (the color of problem triggers depends on the problem severity color, which can be adjusted in the **problem update** screen). Note that recent trigger changes (within the last 2 minutes) will be displayed as blinking blocks.
- Blue up and down arrows indicate triggers that have dependencies. On mouseover, dependency details are revealed.
- A checkbox icon indicates acknowledged problems. All problems or resolved problems of the trigger must be acknowledged for this icon to be displayed.

Clicking on a trigger block provides context-dependent links to problem events of the trigger, the problem acknowledgment screen, trigger configuration, trigger URL or a simple graph/latest values list.

Note that 50 records are displayed by default (configurable in Administration → General → **GUI**, using the Max number of columns and rows in overview tables option). If more records exist than are configured to display, a message is displayed at the bottom of the table, asking to provide more specific filtering criteria. There is no pagination.

Configuration

To configure, select Trigger overview as type:

Add widget
✕

Type
Trigger overview

Show header
☒

Name
default

Refresh interval
Default (1 minute)

Show
Recent problems
Problems
Any

Host groups
type here to search
Select

Hosts
type here to search
Select

Tags
And/Or
Or

tag
Contains
value
Remove

Add

Show suppressed problems
☐

Hosts location
Left
Top

Add
Cancel

In addition to the parameters that are **common** for all widgets, you may set the following specific options:

Show	<p>Filter by problem status:</p> <p>Recent problems - unresolved and recently resolved problems are displayed (default)</p> <p>Problems - unresolved problems are displayed</p> <p>Any - history of all events is displayed</p>
Host groups	<p>Select the host group(s). This field is auto-complete so starting to type the name of a group will offer a dropdown of matching groups.</p>
Hosts	<p>Select hosts. This field is auto-complete so starting to type the name of a host will offer a dropdown of matching hosts. Scroll down to select. Click on 'x' to remove the selected.</p>

Tags

Specify tags to limit the number of item and trigger data displayed in the widget. It is possible to include as well as exclude specific tags and tag values. Several conditions can be set. Tag name matching is always case-sensitive. There are several operators available for each condition:

- Exists** - include the specified tag names
- Equals** - include the specified tag names and values (case-sensitive)
- Contains** - include the specified tag names where the tag values contain the entered string (substring match, case-insensitive)
- Does not exist** - exclude the specified tag names
- Does not equal** - exclude the specified tag names and values (case-sensitive)
- Does not contain** - exclude the specified tag names where the tag values contain the entered string (substring match, case-insensitive)

There are two calculation types for conditions:

- And/Or** - all conditions must be met, conditions having the same tag name will be grouped by the Or condition
- Or** - enough if one condition is met

Mark the checkbox to display problems that would otherwise be suppressed (not shown) because of host maintenance.

Select host location - left or top.

Show suppressed problems

Hosts location

22 URL

Overview

This widget displays the content retrieved from the specified URL.

Configuration

To configure, select URL as type:

Add widget

Type

URL

Name

URL

Refresh interval

Default (No refresh)

* URL

http://

Dynamic item

☐

Add

Cancel

Show header ☒

In addition to the parameters that are **common** for all widgets, you may set the following specific options:

URL

Enter the URL to display.
Relative paths are allowed since Zabbix 4.4.8.
{HOST.*} macros are supported.

Dynamic item

Set to display different URL content depending on the selected host.
This can work if {HOST.*} macros are used in the URL.

Attention:

Browsers might not load an HTTP page included in the widget if Zabbix frontend is accessed over HTTPS.

23 Monitoramento Web

Overview

This widget displays a status summary of the active web monitoring scenarios.

Configuration

Add widget

TypeWeb monitoring

Show header

NameWeb monitoring

Refresh intervalDefault (1 minute)

Host groupstype here to search

Select

Exclude host groupstype here to search

Select

Hoststype here to search

Select

Tags

And/Or

Or

tag

Contains

value

Remove

Add

Show hosts in maintenance

Add

Cancel

Note:

In cases when a user does not have permission to access certain widget elements, that element's name will appear as Inaccessible during the widget's configuration. This results in Inaccessible Item, Inaccessible Host, Inaccessible Group, Inaccessible Map, and Inaccessible Graph appearing instead of the "real" name of the element.

In addition to the parameters that are common for all widgets, you may set the following specific options:

Parameter	Description
Host groups	<p>Enter host groups to display in the widget. This field is auto-complete so starting to type the name of a group will offer a dropdown of matching groups.</p> <p>Specifying a parent host group implicitly selects all nested host groups.</p> <p>Host data from these host groups will be displayed in the widget. If no host groups are entered, all host groups will be displayed.</p>
Exclude host groups	<p>Enter host groups to hide from the widget. This field is auto-complete so starting to type the name of a group will offer a dropdown of matching groups.</p> <p>Specifying a parent host group implicitly selects all nested host groups.</p> <p>Host data from these host groups will not be displayed in the widget. For example, hosts 001, 002, 003 may be in Group A and hosts 002, 003 in Group B as well. If we select to show Group A and exclude Group B at the same time, only data from host 001 will be displayed in the Dashboard.</p>
Hosts	<p>Enter hosts to display in the widget. This field is auto-complete so starting to type the name of a host will offer a dropdown of matching hosts.</p> <p>If no hosts are entered, all hosts will be displayed.</p>
Tags	<p>Specify tags to limit the number of web scenarios displayed in the widget. It is possible to include as well as exclude specific tags and tag values. Several conditions can be set. Tag name matching is always case-sensitive.</p> <p>There are several operators available for each condition:</p> <p>Exists - include the specified tag names</p> <p>Equals - include the specified tag names and values (case-sensitive)</p> <p>Contains - include the specified tag names where the tag values contain the entered string (substring match, case-insensitive)</p> <p>Does not exist - exclude the specified tag names</p> <p>Does not equal - exclude the specified tag names and values (case-sensitive)</p> <p>Does not contain - exclude the specified tag names where the tag values contain the entered string (substring match, case-insensitive)</p> <p>There are two calculation types for conditions:</p> <p>And/Or - all conditions must be met, conditions having the same tag name will be grouped by the Or condition</p> <p>Or - enough if one condition is met</p>
Show hosts in maintenance	<p>Include hosts that are in maintenance in the statistics.</p>

Web monitoring widget

Once you have completed the configuration, you might like to see the widget with the data it displays. To do it, go to Monitoring → Dashboards, click on the name of a dashboard where you created the widget.

In this example, you can see the widget named "Zabbix frontend" displaying the status of the web monitoring for three host groups: "Internal network," "Linux servers," and "Web servers."

Zabbix frontend

Host group ▲ Ok Failed Unknown

Internal network

1

Linux servers

1

Web servers

1

A web monitoring widget displays the following information:

- a name of a widget; below it, there are four columns:
 - Host group - displays a list of host groups that contain hosts having web scenarios configured;
 - Ok - displays a number of web scenarios (in green color) when two conditions are observed:
 - * Zabbix has collected the latest data for a web scenario(s);
 - * all steps that were configured in a **web scenario** are in "Ok" **Status**.
 - Failed - displays a number of web scenarios (in red color), which have some failed steps:
 - * click on the host name, and it will open a new window; the Status column provides detailed information (in red color) on the step where Zabbix failed to collect the data; and also,
 - * gives a hint for the parameter that has to be corrected in the **configuration form**.

Web monitoring					
Host	Name ▲	Number of steps	Last check	Status	Tags
Internal documentation	Internal Wiki	2	38s	Step "Configuration page" [2 of 2] failed: required pattern "winter" was not found on http://localhost/index.php	
Displaying 1 of 1 found					

- Unknown - displays a number of web scenarios (in grey color) for which Zabbix has neither collected data, nor has an information about the failed steps.

Web monitoring					
Host	Name ▲	Number of steps	Last check	Status	Tags
Zabbix site	Zabbix site	1			
Displaying 1 of 1 found					

Viewing the status and data

Clickable links in the widget allow to easily navigate and quickly acquire a full information on each web scenario. Thus, to view:

- the **Status** of a web scenario, click on the name of a host group.
- more detailed statistics, click on the scenario name. In this example it is "Zabbix frontend".
- the details in the case of Failed status, click on a host group name; in the window that opens, click on a web scenario name in the Name column; it will open more detailed information on the configured steps for which Zabbix failed to collect the data.

Details of web scenario: Internal Wiki				
Step	Speed	Response time	Response code	Status
First page	95.94 KBps	256.75ms	200	OK
Configuration page	40.46 KBps	33.5ms	200	Error: required pattern "winter" was not found on http://localhost/index.php
TOTAL		290.25ms		Error: required pattern "winter" was not found on http://localhost/index.php

Now, you can return to the **web scenario configuration form** and correct your settings.

To view the details in the case of Unknown status, you can repeat the same steps as explained for Failed.

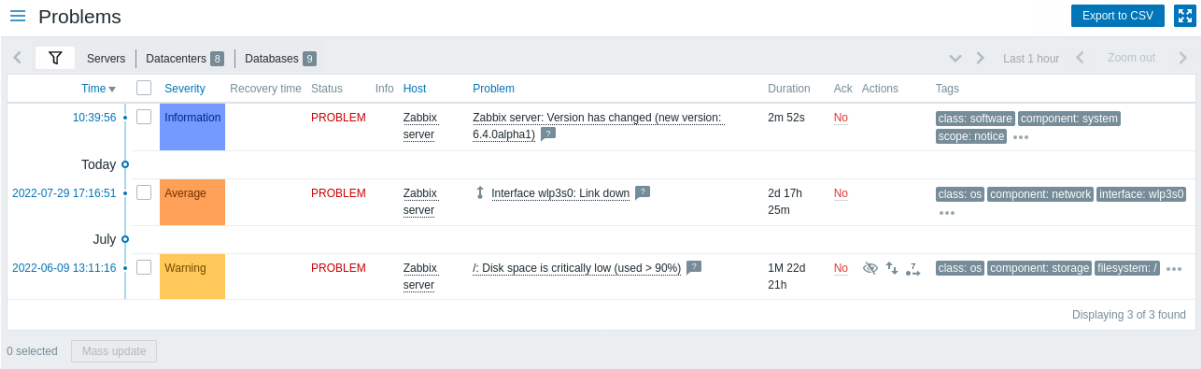
Attention:

At the first monitoring instance, a web scenario is always displayed in Unknown state, which is switched to Failed or Ok state right after the first check. In the case when a host is monitored by the proxy, the status change occurs in accordance with the data collection frequency configured on the proxy.









2 Problems

Overview

In Monitoring → Problems you can see what problems you currently have. Problems are those triggers that are in the "Problem" state.



Column	Description
Time	Problem start time is displayed.
Severity	Problem severity is displayed. Problem severity is originally based on the severity of the underlying problem trigger, however, after the event has happened it can be updated using the Update problem screen. Color of the problem severity is used as cell background during problem time.
Recovery time	Problem resolution time is displayed.
Status	Problem status is displayed: Problem - unresolved problem Resolved - recently resolved problem. You can hide recently resolved problems using the filter. New and recently resolved problems blink for 2 minutes. Resolved problems are displayed for 5 minutes in total. Both of these values are configurable in Administration → General → Trigger displaying options.
Info	A green information icon is displayed if a problem is closed by global correlation or manually when updating the problem. Rolling a mouse over the icon will display more details:
Host	The following icon is displayed if a suppressed problem is being shown (see Show suppressed problems option in the filter). Rolling a mouse over the icon will display more details:

Column	Description
Problem	<p>Problem name is displayed.</p> <p>Problem name is based on the name of the underlying problem trigger.</p> <p>Macros in the trigger name are resolved at the time of the problem happening and the resolved values do not update any more.</p> <p>Note that it is possible to append the problem name with operational data showing some latest item values.</p> <p>Clicking on the problem name brings up the event menu.</p> <p>Hovering on the  icon after the problem name will bring up the trigger description (for those problems that have it).</p> <p>(Note that resolved values of {ITEM.VALUE} and {ITEM.LASTVALUE} macros in trigger descriptions are truncated to 20 characters. To see the entire values you may use macro functions with these macros, e.g. {{ITEM.VALUE}.regsub("(.*)", \1)}, {{ITEM.LASTVALUE}.regsub("(.*)", \1)} as a workaround.)</p>
Operational data	<p>Operational data are displayed containing latest item values.</p> <p>Operational data can be a combination of text and item value macros if configured on a trigger level. If no operational data is configured on a trigger level, the latest values of all items from the expression are displayed.</p> <p>This column is only displayed if Separately is selected for Show operational data in the filter.</p>
Duration	<p>Problem duration is displayed.</p> <p>See also: Negative problem duration</p>
Ack	<p>The acknowledgment status of the problem is displayed:</p> <p>Yes - green text indicating that the problem is acknowledged. A problem is considered to be acknowledged if all events for it are acknowledged.</p> <p>No - a red link indicating unacknowledged events.</p> <p>If you click on the link you will be taken to the problem update screen where various actions can be taken on the problem, including commenting and acknowledging the problem.</p>
Actions	<p>History of activities about the problem is displayed using symbolic icons:</p> <p> - comments have been made. The number of comments is also displayed.</p> <p> - problem severity has been increased (e.g. Information → Warning)</p> <p> - problem severity has been decreased (e.g. Warning → Information)</p> <p> - problem severity has been changed, but returned to the original level (e.g. Warning → Information → Warning)</p> <p> - actions have been taken. The number of actions is also displayed.</p> <p> - actions have been taken, at least one is in progress. The number of actions is also displayed.</p> <p> - actions have been taken, at least one has failed. The number of actions is also displayed.</p> <p>When rolling the mouse over the icons, popups with details about the activity are displayed. See viewing details to learn more about icons used in the popup for actions taken.</p>
Tags	<p>Tags are displayed (if any).</p> <p>In addition, tags from an external ticketing system may also be displayed (see the Process tags option when configuring webhooks).</p>

Operational data of problems

It is possible to display operational data for current problems, i.e. the latest item values as opposed to the item values at the time of the problem.

Operational data display can be configured in the filter of Monitoring → Problems or in the configuration of the respective **dashboard widget**, by selecting one of the three options:

- None - no operational data is displayed
- Separately - operational data is displayed in a separate column

Time	<input type="checkbox"/>	Severity	Recovery time	Status	Info	Host	Problem	Operational data	Duration
09:28:35	<input type="checkbox"/>	Average		PROBLEM		Zabbix server	Zabbix discoverer processes more than 75% busy	Current value: 100 %	3h 32m 8s

- With problem name - operational data is appended to the problem name and in parentheses. Operational data are appended to the problem name only if the Operational data field is non-empty in the trigger configuration.

Time	<input type="checkbox"/>	Severity	Recovery time	Status	Info	Host	Problem	Duration
09:28:35	<input type="checkbox"/>	Average		PROBLEM		Zabbix server	Zabbix discoverer processes more than 75% busy (Current value: 100 %)	3h 29m 34s

The content of operational data can be configured with each **trigger**, in the Operational data field. This field accepts an arbitrary string with macros, most importantly, the `{ITEM.LASTVALUE<1-9>}` macro.

`{ITEM.LASTVALUE<1-9>}` in this field will always resolve to the latest values of items in the trigger expression. `{ITEM.VALUE<1-9>}` in this field will resolve to the item values at the moment of trigger status change (i.e. change into problem, change into OK, being closed manually by a user or being closed by correlation).

Negative problem duration

It is actually possible in some common situations to have negative problem duration i.e. when the problem resolution time is earlier than problem creation time, e. g.:

- If some host is monitored by proxy and a network error happens, leading to no data received from the proxy for a while, the `nodata(/host/key)` trigger will be fired by the server. When the connection is restored, the server will receive item data from the proxy having a time from the past. Then, the `nodata(/host/key)` problem will be resolved and it will have a negative problem duration;
- When item data that resolve the problem event are sent by Zabbix sender and contain a timestamp earlier than the problem creation time, a negative problem duration will also be displayed.

Note:

Negative problem duration is not affecting **SLA calculation** or **Availability report** of a particular trigger in any way; it neither reduces nor expands problem time.

Mass editing options

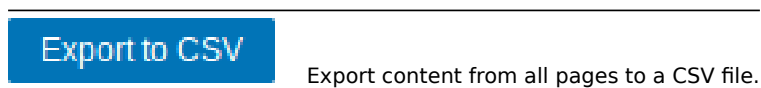
Buttons below the list offer some mass-editing options:

- Mass update - update the selected problems by navigating to the **problem update** screen

To use this option, mark the checkboxes before the respective problems, then click on the Mass update button.

Buttons

The button to the right offers the following option:



View mode buttons, being common for all sections, are described on the **Monitoring** page.

Using filter

You can use the filter to display only the problems you are interested in. For better search performance, data is searched with macros unresolved.

The filter is located above the table. Favorite filter settings can be saved as tabs and then quickly accessed by clicking on the **tabs above the filter**.

Parameter	Description
Show	<p>Filter by problem status:</p> <p>Recent problems - unresolved and recently resolved problems are displayed (default)</p> <p>Problems - unresolved problems are displayed</p> <p>History - history of all events is displayed</p>
Host groups	<p>Filter by one or more host groups.</p> <p>Specifying a parent host group implicitly selects all nested host groups.</p>
Hosts	Filter by one or more hosts.
Triggers	Filter by one or more triggers.
Problem	Filter by problem name.
Severity	Filter by trigger (problem) severity.
Age less than	Filter by how old the problem is.
Host inventory	Filter by inventory type and value.
Tags	<p>Filter by event tag name and value. It is possible to include as well as exclude specific tags and tag values. Several conditions can be set. Tag name matching is always case-sensitive.</p> <p>There are several operators available for each condition:</p> <p>Exists - include the specified tag names</p> <p>Equals - include the specified tag names and values (case-sensitive)</p> <p>Contains - include the specified tag names where the tag values contain the entered string (substring match, case-insensitive)</p> <p>Does not exist - exclude the specified tag names</p> <p>Does not equal - exclude the specified tag names and values (case-sensitive)</p> <p>Does not contain - exclude the specified tag names where the tag values contain the entered string (substring match, case-insensitive)</p> <p>There are two calculation types for conditions:</p> <p>And/Or - all conditions must be met, conditions having the same tag name will be grouped by the Or condition</p> <p>Or - enough if one condition is met</p> <p>When filtered, the tags specified here will be displayed first with the problem, unless overridden by the Tag display priority (see below) list.</p> <p>Select the number of displayed tags:</p> <p>None - no Tags column in Monitoring → Problems</p> <p>1 - Tags column contains one tag</p> <p>2 - Tags column contains two tags</p> <p>3 - Tags column contains three tags</p> <p>To see all tags for the problem roll your mouse over the three dots icon.</p>
Show tags	

Parameter	Description
Tag name	Select tag name display mode: Full - tag names and values are displayed in full Shortened - tag names are shortened to 3 symbols; tag values are displayed in full None - only tag values are displayed; no names
Tag display priority	Enter tag display priority for a problem, as a comma-separated list of tags (for example: <code>Services,Applications,Application</code>). Tag names only should be used, no values. The tags of this list will always be displayed first, overriding the natural ordering by alphabet.
Show operational data	Select the mode for displaying operational data : None - no operational data is displayed Separately - operational data is displayed in a separate column With problem name - append operational data to the problem name, using parentheses for the operational data
Show suppressed problems	Mark the checkbox to display problems that would otherwise be suppressed (not shown) because of host maintenance.
Compact view	Mark the checkbox to enable compact view.
Show details	Mark the checkbox to display underlying trigger expressions of the problems. Disabled if Compact view checkbox is marked.
Show unacknowledged only	Mark the checkbox to display unacknowledged problems only.
Show timeline	Mark the checkbox to display the visual timeline and grouping. Disabled if Compact view checkbox is marked.
Highlight whole row	Mark the checkbox to highlight the full line for unresolved problems. The problem severity color is used for highlighting. Enabled only if the Compact view checkbox is marked in the standard blue and dark themes. Highlight whole row is not available in the high-contrast themes.

Abas para filtros favoritos

Conjuntos de parâmetros de filtro usados com frequência podem ser salvos em guias.

Para salvar um novo conjunto de parâmetros de filtro, abra a guia principal e configure as configurações do filtro e, em seguida, pressione o botão Salvar como. Em um novo pop-up janela, defina Propriedades do filtro.

Filter properties

* Name

Server problems

Show number of records

☒

Set custom time period

☒

From

now-1d

To

now

Delete

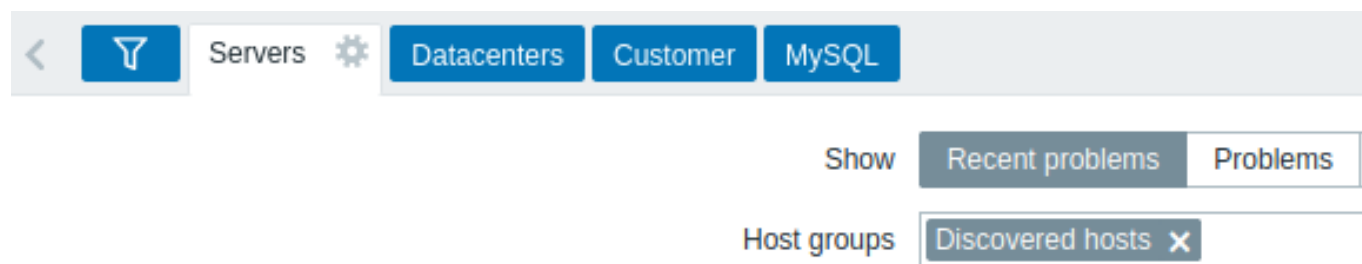
Save

Cancel

Parâmetro	Descrição
Nome	O nome do filtro a ser exibido na lista de guias.
Mostrar número de registros	Marque, se desejar que o número de problemas seja exibido ao lado do nome da guia.

Parâmetro	Descrição
Definir período de tempo personalizado	Marque para definir o período de tempo padrão específico para este conjunto de filtros. Se definido, você só poderá alterar o período de tempo dessa guia atualizando as configurações do filtro. Para guias sem um período de tempo personalizado, o intervalo de tempo pode ser alterado pressionando o botão seletor de tempo no canto superior direito (o nome do botão depende do intervalo de tempo selecionado: Esta semana, Últimos 30 minutos, Ontem etc.). Esta opção está disponível apenas para filtros em Monitoramento→Problemas.
De/Para	Período de tempo começam e terminam em sintaxe de tempo absoluta (Y-m-d H:i:s) ou relativa (now-1d). Disponível, se Definir período de tempo personalizado estiver marcado.

Para editar Propriedades do filtro de um filtro existente, pressione o símbolo de engrenagem ao lado do nome da guia ativa.



Notas:

- Para ocultar um filtro, pressione o nome da aba atual. Pressione no nome da guia ativa novamente para abrir o filtro.
- As guias de filtro podem ser reorganizadas arrastando e soltando.
- A navegação do teclado é suportada: use as setas para alternar entre as guias, pressione Enter para abrir.
- Pressionar o ícone de seta para baixo no canto superior direito abrirá o lista completa de guias de filtro salvas como um menu suspenso.

Note:

Para compartilhar filtros, copie e envie para outras pessoas um URL de um filtro ativo. Depois de abrir este URL, outros usuários poderão salvar este conjunto de parâmetros como um filtro permanente em sua conta Zabbix.

Veja também: [Página parâmetros](#).

Botões de filtro

Apply	Aplicar critérios de filtragem especificados (sem salvar).
Reset	Redefinir o filtro atual e retornar aos parâmetros salvos da guia atual . Na guia principal, isso limpará o filtro.
Save as	Salvar parâmetros de filtro atuais em uma nova guia. Disponível apenas na guia principal.
Update	Substitua os parâmetros da guia pelos parâmetros atualmente especificados. Não disponível na guia principal.

Event menu

Clicking on the problem name brings up the event menu:

Problem

/: Disk space is critically low (>90% used)

TRIGGER

Problems

Configuration

HISTORY

Trap item (text)

SCRIPTS

Check disk space A

Check disk space S

Detect operating system

New webhook

Ping

The event menu allows to:

- filter the problems of the trigger
- access the trigger configuration
- access a simple graph/item history of the underlying item(s)
- access an external ticket of the problem (if configured, see the Include event menu entry option when configuring **webhooks**)
- execute global **scripts** (these scripts need to have their scope defined as 'Manual event action'). This feature may be handy for running scripts used for managing problem tickets in external systems.

Viewing details












The times for problem start and recovery in Monitoring → Problems are links. Clicking on them opens more details of the event.

Event details

Trigger details		Actions					
Host	New host	Step	Time	User/Recipient	Action	Message/Command	Status Info
Trigger	CPU load too high on "New host" for 3 minutes		2019-10-15 16:18:04	Admin (Zabbix Administrator)	✓		
Severity	Warning		2019-10-15 16:17:42	Admin (Zabbix Administrator)	✕ +	OK.	
Problem expression	{New host:system.cpu.load.avg(3m)}>2	1	2019-10-15 16:12:35	Admin (Zabbix Administrator)	✉	Problem: CPU load too high on "New host" for 3 minutes	Sent
Recovery expression				@inbox.lv		Problem started at 16:12:35 on 2019.10.15 Problem name: CPU load too high on "New host" for 3 minutes Host: New host Severity: Not classified Original problem ID: 295677	
Event generation	Normal		2019-10-15 16:12:35		📅		
Allow manual close	No	Event list [previous 20]					
Enabled	Yes	Time	Recovery time	Status	Age	Duration	Ack Actions
Event details		2019-10-15 16:12:35		PROBLEM	7m 29s	7m 29s	Yes 🗨️ ⬇️ ⚙️
Event	CPU load too high on "New host" for 3 minutes	2019-10-15 15:10:05	2019-10-15 16:08:35	RESOLVED	1h 9m 59s	58m 30s	No
Operational data	1.99	2019-10-15 14:58:05	2019-10-15 15:08:35	RESOLVED	1h 21m 59s	10m 30s	No
Severity	Information	2019-10-15 14:50:35	2019-10-15 14:54:35	RESOLVED	1h 29m 29s	4m	No
Time	2019-10-15 16:12:35	2019-10-15 13:14:05	2019-10-15 13:25:35	RESOLVED	3h 5m 59s	11m 30s	No
Acknowledged	Yes	2019-10-15 13:02:05	2019-10-15 13:08:35	RESOLVED	3h 17m 59s	6m 30s	No
Tags	Service: Operations						
Description							

Note how the problem severity differs for the trigger and the problem event - for the problem event it has been updated using the Update problem **screen**.

In the action list, the following icons are used to denote the activity type:

-  - problem event generated
-  - message has been sent
-  - problem event acknowledged
-  - problem event unacknowledged
-  - a comment has been added
-  - problem severity has been increased (e.g. Information → Warning)
-  - problem severity has been decreased (e.g. Warning → Information)
-  - problem severity has been changed, but returned to the original level (e.g. Warning → Information → Warning)
-  - a remote command has been executed
-  - problem event has recovered
-  - the problem has been closed manually


3 Hosts

Visão geral

A seção Monitoramento → Hosts exibe uma lista completa de hosts monitorados com informações detalhadas sobre a interface do host, disponibilidade, tags, problemas atuais, status (ativado/desativado) e links para navegue até os dados mais recentes do host, histórico de problemas, gráficos, painéis e cenários da web.

Hosts Create host

Name	Interface	Availability	Tags	Status	Latest data	Problems	Graphs	Dashboards	Web
Apache server DC1	127.0.0.1:10050	ZBX		Enabled	Latest data	Problems	Graphs	Dashboards	Web
Zabbix NYC	127.0.0.1:10050	ZBX	Apache	Enabled	Latest data 2	1	Graphs 27	Dashboards 3	Web
Zabbix server	127.0.0.1:10050	ZBX		Enabled	Latest data 163	1 2 1 1	Graphs 27	Dashboards 3	Web
Zabbix Tokyo	127.0.0.1:10050	ZBX		Enabled	Latest data 26	1	Graphs 5	Dashboards 2	Web

Coluna	Descrição
Nome	O nome do host visível. Clicar no nome abre o menu do host . Um ícone de chave laranja  após o nome indica que este host está em manutenção.
Interface	Clique no cabeçalho da coluna para classificar os hosts por nome em ordem crescente ou decrescente. A interface principal do host é exibida.
Disponibilidade	Host availability por interface configurada. Os ícones representam apenas os tipos de interface (agente Zabbix, SNMP, IPMI, JMX) que estão configurados. Se você posicionar o mouse sobre o ícone, uma lista pop-up de todas as interfaces desse tipo aparecerá com os detalhes, status e erros de cada interface. A coluna está vazia para hosts sem interfaces. O status atual de todas as interfaces de um tipo é exibido pela respectiva cor do ícone: Verde - todas as interfaces disponíveis Amarelo - pelo menos uma interface disponível e pelo menos uma indisponível; outros podem ter qualquer valor, incluindo 'unknown' Red - nenhuma interface disponível Gray - pelo menos uma interface desconhecida (nenhuma indisponível) Observe que os itens ativos do agente Zabbix não afetar a disponibilidade do host.
Tags	Tags do host e todos os modelos vinculados, com macros não resolvidas.

Coluna	Descrição
Status	Status do host - Ativado ou Desativado. Clique no cabeçalho da coluna para classificar os hosts por status em ordem crescente ou decrescente.
Últimos dados	Clicar no link abrirá a página Monitoramento - Últimos dados com todos os dados mais recentes coletados do host.
Problemas	O número de problemas de host abertos classificados por gravidade. A cor do quadrado indica a gravidade do problema. O número no quadrado significa o número de problemas para determinada gravidade. Clicar no ícone abrirá a página Monitoramento - Problemas para o host atual. Se um host não tiver problemas, um link para a seção Problemas desse host será exibido como texto. Use o filtro para selecionar se os problemas suprimidos devem ser incluídos (não incluídos por padrão).
Gráficos	Clicar no link exibirá os gráficos configurados para o host. O número de gráficos é exibido em cinza. Se um host não possui gráficos, o link está desabilitado (texto cinza) e nenhum número é exibido.
Painéis	Clicar no link exibirá os painéis configurados para o host. O número de painéis é exibido em cinza. Se um host não tiver painéis, o link será desabilitado (texto cinza) e nenhum número será exibido.
Web	Clicar no link exibirá cenários da web configurados para o host. O número de cenários da web é exibido em cinza. Se um host não tiver cenários da web, o link será desabilitado (texto cinza) e nenhum número será exibido.

Botões

Criar host permite criar um **novo host**. Este botão está disponível apenas para usuários Admin e Super Admin.

Os botões do modo de visualização comuns a todas as seções estão descritos na **Monitoramento** página.

Usando filtro

Você pode usar o filtro para exibir apenas os hosts nos quais está interessado. Para melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O filtro está localizado acima da tabela. É possível filtrar hosts por nome, grupo de hosts, IP ou DNS, porta de interface, tags, gravidade do problema, status (ativado/desativado/qualquer); você também pode selecionar se deseja exibir problemas suprimidos e hosts que estão atualmente em manutenção. As configurações de filtro favoritas podem ser salvas como guias e acessadas rapidamente clicando nas **guias acima da filter**.

Parâmetro	Descrição
Nome	Filtrar por nome de host visível.
Grupos de hosts	Filtrar por um ou mais grupos de hosts. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
IP	Filtrar por endereço IP.
DNS	Filtrar por nome DNS.
Porta	Filtrar por número de porta.

Os gráficos são classificados por:

- nome do gráfico (gráficos personalizados)
- nome do item (gráficos simples)

Gráficos para hosts desabilitados também são acessíveis.

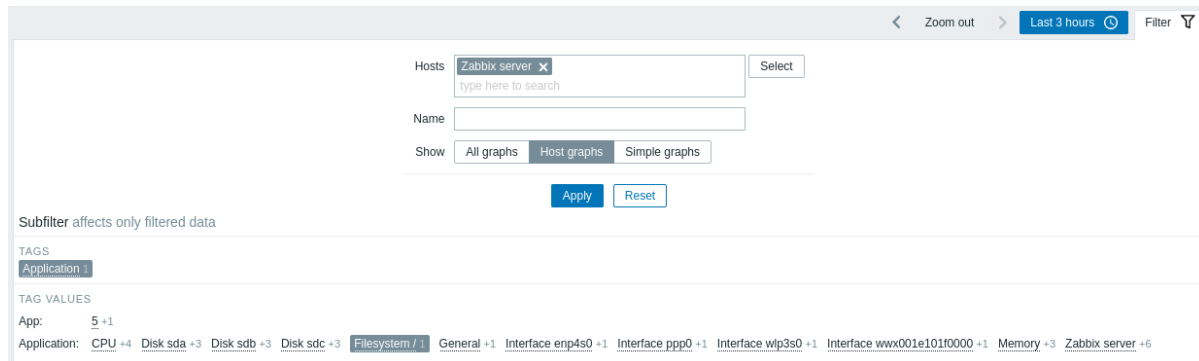
Seletor de período de tempo

Anote o seletor de período de tempo acima do gráfico. Permite selecionando períodos frequentemente necessários com um clique do mouse.

Veja também: [Período de tempo seletor](#)

Usando filtro

Para visualizar um gráfico específico, selecione-o no filtro. O filtro permite especificar o host, o nome do gráfico e a opção Show (todos/gráficos do host/gráficos simples).



Se nenhum host for selecionado no filtro, nenhum gráfico será exibido.

Usando subfiltro

O subfiltro é útil para um acesso rápido com um clique aos gráficos relacionados. O subfiltro opera de forma autônoma a partir do filtro principal - os resultados são filtrados imediatamente, sem necessidade de clicar em Aplicar no filtro principal.

Observe que o subfiltro só permite modificar ainda mais a filtragem do filtro principal.

Ao contrário do filtro principal, o subfiltro é atualizado junto com cada solicitação de atualização da tabela para obter sempre informações atualizadas das opções de filtragem disponíveis e seu contador números.

O subfiltro mostra **links clicáveis** que permitem filtrar gráficos com base em uma entidade - o nome da tag ou o valor da tag. Assim que a entidade é clicada, os gráficos são filtrados imediatamente; a entidade selecionada é destacada com fundo cinza. Para remover a filtragem, clique novamente na entidade. Para adicionar outra entidade aos resultados filtrados, clique em outra entidade.

O número de entidades exibidas é limitado a 100 horizontalmente. Se houver mais, um ícone de três pontos é exibido no final; não é clicável. Listas verticais (como tags com seus valores) são limitadas a 20 entradas. Se houver mais, um ícone de três pontos é exibido; não é clicável.

Um número ao lado de cada entidade clicável indica o número de gráficos que ela possui nos resultados do filtro principal.

Uma vez que uma entidade é selecionada, os números com outras entidades disponíveis são exibidos com um sinal de mais indicando quantos gráficos podem ser adicionados à seleção atual.

Botões

Os botões do modo de visualização, comuns a todas as seções, estão descritos na [Monitoramento](#) página.

2 Cenários Web

Visão geral

As informações do host [cenário da web](#) podem ser acessadas de Monitoring → Hosts clicando na Web para o respectivo host.

Filter

Host	Name ▲	Number of steps	Last check	Status	Tags
New host	Zabbix frontend	5	46s	OK	Application: Zabbix fro...

Displaying 1 of 1 found

Os dados de hosts desativados também estão acessíveis. O nome de um host desativado está listado em vermelho.

O número máximo de cenários exibidos por página depende das Rows por página perfil de usuário [configuração](#).

Por padrão, apenas os valores que se enquadram nas últimas 24 horas são exibido. Este limite foi introduzido com o objectivo de melhorar a tempos de carregamento iniciais para páginas grandes de dados mais recentes. Você pode estender este período alterando o valor de Período máximo de exibição do histórico parâmetro no [Administração→Geral](#) seção do cardápio.

O nome do cenário é um link para estatísticas mais detalhadas sobre ele:

Details of web scenario: Zabbix frontend



Usando filtro

A página mostra uma lista de todos os cenários da web do host selecionado. Ver cenários da web para outro host ou grupo de hosts sem retornar ao Monitoramento → Hosts, selecione esse host ou grupo no filtro. Você também pode filtrar cenários com base em tags.

Botões

Os botões do modo de visualização comuns a todas as seções estão descritos na [Monitoramento](#) página.

4 Dados recentes

Visão geral

Nesta seção você pode ver os valores mais recentes reunidos por itens.

Os gráficos também estão disponíveis para os valores dos itens.

Latest data

Subfilter affects only filtered data

HOSTS

Zabbix server 2

TAG VALUES

Application: Interface enp4s0 Interface ppp0 +2 Interface wlp3s0 +2

<input type="checkbox"/> Host	Name ▲	Last check	Last value	Change	Tags	Info
<input type="checkbox"/> Zabbix server	Interface enp4s0: Bits received	3s	5.35 Kbps	-496 bps	Application: Interface ...	Graph
<input type="checkbox"/> Zabbix server	Interface enp4s0: Bits sent	3s	992 bps	-144 bps	Application: Interface ...	Graph

0 selected Display stacked graph Display graph Execute now

Displaying 2 of 2 found

Esta seção contém:


- o **filtro** (recolhido por padrão)
- o **subfiltro** (nunca recolhido)
- a lista de itens


Os itens são exibidos com seu nome, tempo desde a última verificação, **último valor**, alteração quantidade, tags e um link para um gráfico/histórico simples de valores de itens.


Clicar no nome do item abre o menu do item com links para gráficos disponíveis e o configuração de itens.

As tags na lista de itens são clicáveis. Se você clicar em uma tag, ela será habilitada no subfiltro. A lista de itens agora exibe os itens correspondentes a esta tag e quaisquer outras tags selecionadas anteriormente no subfiltro. Observe que, uma vez que os itens tenham foram filtradas dessa forma, as tags na lista não serão mais clicáveis. Modificação adicional com base em tags (por exemplo, remover, adicionar outro filtro) deve ser feito no subfiltro.

Os valores na coluna Último valor são exibidos com conversão de unidade e mapeamento de valor aplicado. Para visualizar dados brutos, passe o mouse sobre o valor.

Se um item tiver erros, por exemplo, se tornou incompatível, um ícone de informação será ser exibido na coluna Info . Passe o mouse sobre o ícone para obter detalhes.

Um ícone com um ponto de interrogação  é exibido ao lado do nome do item para todos os itens que têm um Descrição. Passe o mouse sobre este ícone para ver uma dica de ferramenta com o item Descrição.

Se um host ao qual o item pertence estiver em manutenção, um ícone de chave laranja  é exibido após o nome do host.

Observação: O nome de um host desativado é exibido em vermelho. Os dados de hosts desativados, incluindo gráficos e listas de valores de itens, também podem ser acessados em Dados mais recentes.

Por padrão, apenas os valores que se enquadram nas últimas 24 horas são exibido. Este limite foi introduzido com o objetivo de melhorar tempos de carregamento iniciais para páginas grandes dos dados mais recentes. Este período de tempo pode ser estendido alterando o valor de o parâmetro Período máximo de exibição do histórico em Administração → [Geral]/(manual/web_interface/frontend_sections/administration/general#gui).

::: não importante Para itens com uma frequência de atualização de 1 dia ou mais o valor da mudança nunca será exibido (com o padrão contexto). Também neste caso o último valor não será exibido se foi recebido há mais de 24 horas. :::

Buttons

View mode buttons being common for all sections are described on the [Monitoring](#) page.

Usando filtro

Você pode usar o filtro para exibir apenas os itens de seu interesse. Para melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O ícone de filtro  está localizado acima da tabela e do subfiltro. Clique nisso para expandir o filtro.

O filtro permite restringir a lista por grupo de hosts, host, nome do item, tag e outras configurações. Especificar um grupo de hosts pai no filtro seleciona implicitamente todos os hosts aninhados grupos. Consulte Monitoramento -> **Problemas** para obter detalhes sobre como filtrar por tags.

Mostrar detalhes permite estender as informações exibidas para os itens. Detalhes como intervalo de atualização, configurações de histórico e tendências, itens tipo e os erros de item (bem/sem suporte) são exibidos.

Salvando filtro

As configurações de filtro podem ser salvas para acesso rápido mais tarde. Para salvar um filtro, clique no botão Salvar como e digite o nome do filtro e a opção de exibir ou ocultar o número de registros no filtro salvo. Quando salvo, o filtro é criado como um guia filtro e imediatamente ativado.

Os botões esquerdo/direito acima do filtro podem ser usados para navegar entre filtros. Alternativamente, o botão apontando para baixo abre um menu suspenso com todos os filtros salvos e você pode clicar no que você precisa.

Se as configurações de um filtro salvo foram alteradas, um asterisco é exibido antes o nome do filtro em itálico. Para atualizar o filtro de acordo com as novas configurações, clique no botão Atualizar, que é exibido em vez de o botão Salvar como.

O botão Reset permite desativar as configurações de filtro não salvas na tela principal filtro e o subfiltro.

Usando subfiltro

O subfiltro é útil para um acesso rápido com um clique a grupos de itens relacionados. O subfiltro opera de forma autônoma a partir do filtro principal - os resultados são filtrados imediatamente, sem necessidade de clicar em Aplicar no filtro principal.

Observe que o subfiltro só permite modificar ainda mais a filtragem do filtro principal.

Ao contrário do filtro principal, o subfiltro é atualizado junto com cada solicitação de atualização da tabela para obter sempre informações atualizadas das opções de filtragem disponíveis e seu contador números.

O subfiltro mostra **links clicáveis** que permitem filtrar itens com base em um entidade - o host, o nome da tag ou o valor da tag. Assim que a entidade é clicada, os itens são filtrado imediatamente; a entidade selecionada é destacada com fundo cinza. Para remover a filtragem, clique novamente na entidade. Para adicionar outro entidade aos resultados filtrados, clique em outra entidade.

O número de entidades exibidas é limitado a 100 horizontalmente. Se houver mais, um ícone de três pontos é exibido no final; não é clicável. Listas verticais (como tags com seus valores) são limitados a 20 entradas. Se houver mais, um ícone de três pontos é exibido; não é clicável.

As opções de host no subfiltro estão disponíveis somente se não houver hosts ou mais de um host é selecionado no filtro principal.

Por padrão, os itens com e sem dados são exibidos na lista de itens. Se apenas um hospedeiro estiver selecionado no filtro principal, o subfiltro oferece a opção de filtrar apenas os itens com dados, apenas sem dados, ou ambos para este host.

Um número ao lado de cada entidade clicável indica o número de itens que ela possui no resultados do filtro principal. Entidades sem itens não são exibidas, a menos que tenham sido selecionado no subfiltro antes.

Uma vez que uma entidade é selecionada, os números com outras entidades disponíveis são exibidos com um sinal de mais indicando quantos itens podem ser adicionados à seleção atual.

Gráficos

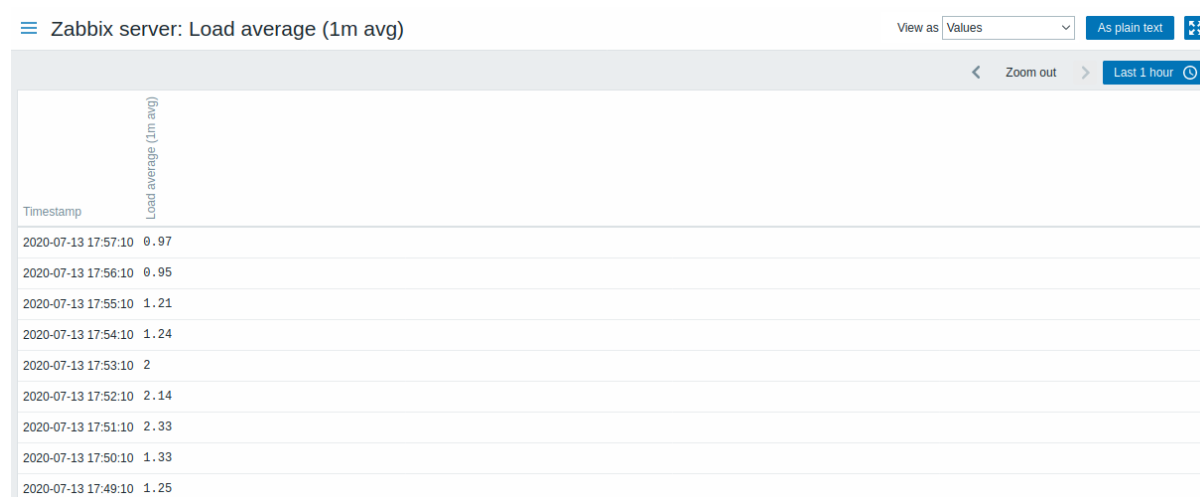
Gráficos ad-hoc para comparar itens

Você pode usar a caixa de seleção na primeira coluna para selecionar vários itens e em seguida, compare seus dados de forma simples ou empilhada **ad-hoc gráfico**. Para isso, selecione itens de interesse e, em seguida, clique no botão gráfico necessário abaixo do tabela.

Links para histórico de valores/gráfico simples

A última coluna na lista de valores mais recente oferece:

- um link **Histórico** (para todos os itens textuais) - levando a listagens (Valores/500 valores mais recentes) exibindo o histórico do item anterior valores.
- um link **Gráfico** (para todos os itens numéricos) - levando a um **simples gráfico**. No entanto, uma vez o gráfico é exibido, uma lista suspensa no canto superior direito oferece uma possibilidade de mudar para Valores/500 valores mais recentes também.



Os valores exibidos nesta lista são "brutos", ou seja, sem pós-processamento é aplicado.

Note:

A quantidade total de valores exibidos é definida pelo valor do parâmetro Limite para resultados de pesquisa e filtro, definido em **Administração → Geral**.

5 Mapas

[comentário]: # (tags: mapa)

Visão geral

Através do módulo Monitoramento → Mapas você pode manter e visualizar os **mapas de rede**.

Este módulo tem a característica de lembrar qual foi o último mapa que você acessou (ou a listagem). Então quando você acessa este módulo ele irá se configurar com o último mapa ou listagem. A partir do Zabbix 3.0 os mapas podem ser públicos ou privados, mapas públicos estão disponíveis para todos os usuários, enquanto os privados somente para seus proprietários e para os usuários com os quais eles foram compartilhados.

Listagem dos mapas

The screenshot shows the 'Maps' list interface in Zabbix. It includes a 'Create map' and 'Import' button at the top right. Below the table, there is a 'Filter' button. The table has the following data:

<input type="checkbox"/> Name ▲	Width	Height	Actions
<input type="checkbox"/> Local network	600	400	Properties Constructor
<input type="checkbox"/> Local network2	680	200	Properties Constructor

At the bottom, it says '0 selected' and 'Export Delete'. On the right, it says 'Displaying 2 of 2 found'.

São apresentadas as seguintes colunas:

Coluna	Descrição
Nome	Nome único do mapa. Ao clicar no nome do mapa você é redirecionado para a visualização do mapa.
Largura	Largura do mapa.
Altura	Altura do mapa.

Coluna	Descrição
Ações	Estão disponíveis duas ações: Propriedades - edita as características gerais (externas) do mapa Construtor - acessa o modo de edição interna dos elementos do mapa

Para **configurar** um novo mapa, clique no botão Criar mapa na barra de título. Para importar um mapa a partir de um arquivo XML, clique no botão Importar situado ao lado.

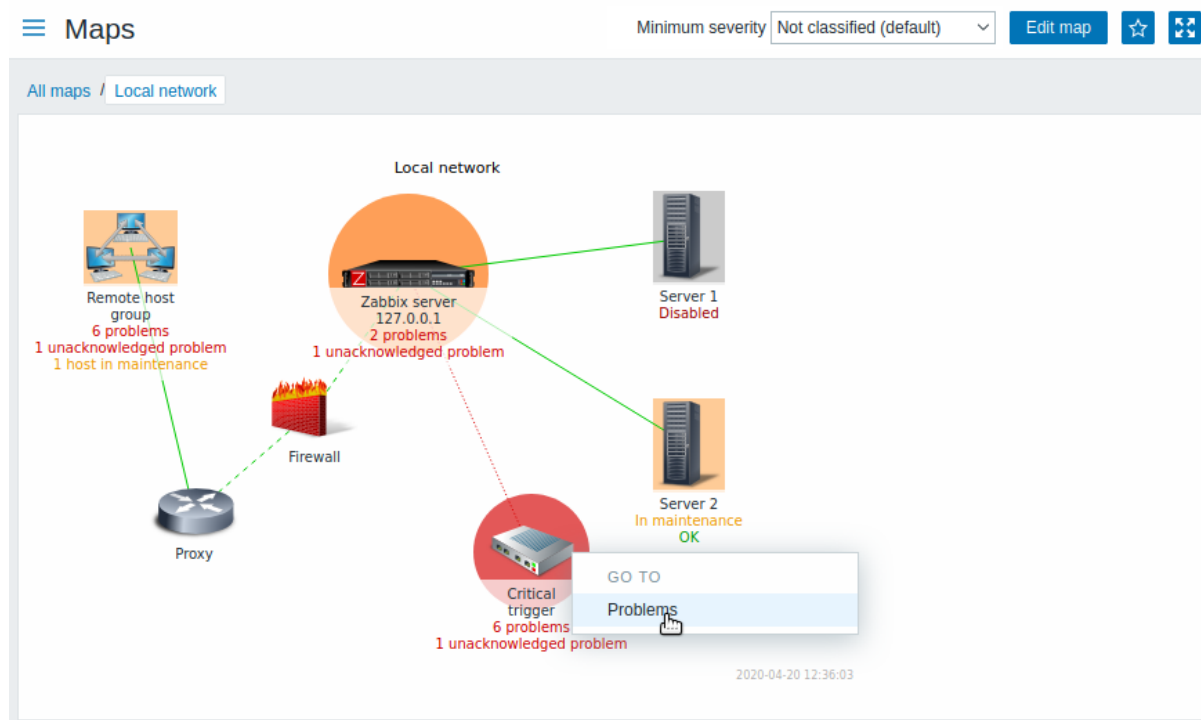
Na parte inferior da lista de mapas existem dois botões para ações em massa:

- Exportar - exporta os mapas selecionados para um arquivo XML
- Excluir - exclui os mapas selecionados

Para usar estas opções, marque os mapas desejados e clique no botão apropriado.

Visualizando os mapas

Para visualizar um mapa, clique em seu nome na lista de mapas.



Você também pode utilizar a caixa de seleção situada na barra de título para selecionar o menor nível de severidade que deverá ser apresentado nos incidentes visualizáveis pelo mapa. A severidade marcada como padrão no mapa será a que foi definida nas características gerais do mesmo. Se o mapa contiver um submapa, a navegação pelo submapa respeitará o maior nível de severidade selecionado.

Realce de ícones

Se um elemento do mapa estiver em estado de incidente, ele terá um círculo atrás de si. A cor deste círculo dependerá do maior nível de severidade de incidente atualmente ativo no host. Apenas incidentes com nível de severidade igual ou superior ao definido para a visualização serão apresentados. Se todos os incidentes para o elemento tiverem sido reconhecidos, o círculo terá uma borda verde.

Adicionalmente um host em **manutenção** será destacado com um uma área quadrada na cor laranja, e hosts com monitoração inativa (não monitorados) serão destacados com um quadrado cinza. O destaque de ícones será apresentado se a opção Realce de ícones tiver sido marcada durante a **configuração do mapa**.

Indicadores de mudança recente



Triângulos vermelhos ao redor de um elemento indicam que este teve alteração recente de status em alguma trigger, nos últimos 30 minutos. Estes triângulos são apresentados se a opção Destacar elementos com modificação de status em trigger tiver sido marcada durante a **configuração do mapa**.

Links

Clicar em elementos dos mapas pode abrir menus flutuantes com links.




Controles

Quando no modo de visualização, na barra de título dos mapas existem dois botões de controle:

-  - adiciona o mapa ao widget de favoritos no **Dashboard**
-  - exibe o mapa em modo de tela cheia

Buttons

Buttons to the right offer the following options:

	Go to map constructor to edit the map content.
	Add map to the favorites widget in the Dashboard .
	The map is in the favorites widget in the Dashboard . Click to remove map from the favorites widget.

View mode buttons being common for all sections are described on the **Monitoring** page.

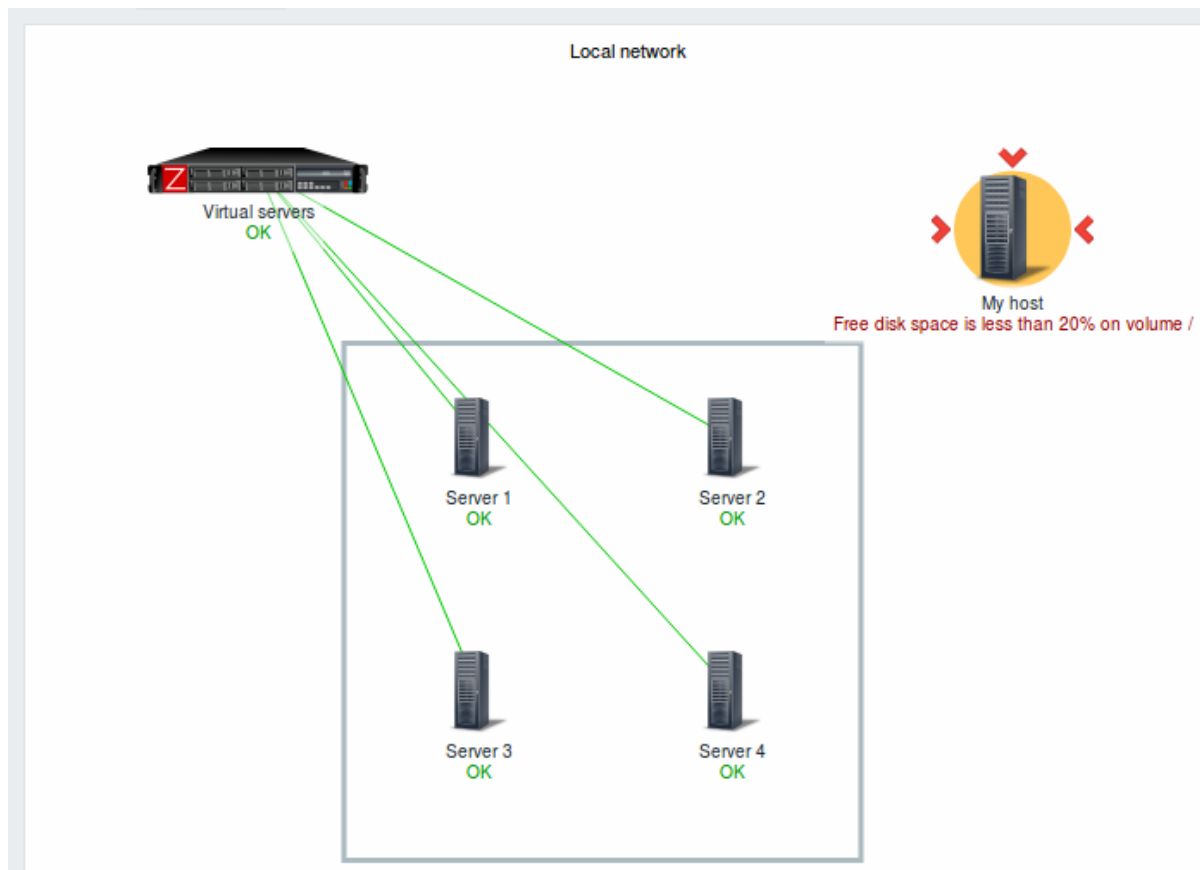
Readable summary in maps

A hidden "aria-label" property is available allowing map information to be read with a screen reader. Both general map description and individual element description is available, in the following format:

- for map description: `<Map name>, <* of * items in problem state>, <* problems in total>.`
- for describing one element with one problem: `<Element type>, Status <Element status>, <Element name>, <Problem description>.`
- for describing one element with multiple problems: `<Element type>, Status <Element status>, <Element name>, <* problems>.`
- for describing one element without problems: `<Element type>, Status <Element status>, <Element name>.`

For example, this description is available:

'Local network, 1 of 6 elements in problem state, 1 problem in total. Host, Status problem, My host, Free
for the following map:



Referencing a network map

Network maps can be referenced by both `sysmapid` and `mapname` GET parameters. For example,

`http://zabbix/zabbix/zabbix.php?action=map.view&mapname=Local%20network`

will open the map with that name (Local network).

If both `sysmapid` (map ID) and `mapname` (map name) are specified, `mapname` has higher priority.

6 Descoberta

Visão geral

Nos resultados da seção Monitoramento → Descoberta de **rede descoberta** são mostrados. Descoberto os dispositivos são classificados pela regra de descoberta.

≡ Status of discovery Filter

Discovery rule Select

Apply Reset

Discovered device ▾	Monitored host	Uptime/Downtime	SNMPv2 agent: iso.3.6.1.2.1.1.0
Local network (14 devices)			
192.168.3.114 (radix-ilo.zabbix.lan)	Integrated Lights-Out 4 2.61 Jul 27 2018	1d 2h 47m	
192.168.3.72 (winxp.zabbix.lan)	Linux zeus 4.8.6.5-smp_2 SMP Sun Nov 13 14_58_11 CDT 2016 i686	7 days, 20:37:53	7d 20h 37m
192.168.3.70 (win2008i386.zabbix.lan)	Hardware_ x86 Family 6 Model 23 Stepping 6 AT_AT COMPATIBLE - Software_ Windows Version 6.0_Build 6001 Multiprocessor Free_	2 days, 02:23:47	2d 2h 23m

Se um dispositivo já estiver monitorado, o nome do host será listado na Coluna Host monitorado e a duração do dispositivo sendo descoberto ou perdido após a descoberta anterior ser mostrado no Uptime/Downtime coluna.

Depois disso, siga as colunas que mostram o estado dos serviços individuais para cada dispositivo descoberto (células vermelhas mostram serviços que estão inativos). O tempo de atividade ou o tempo de inatividade do serviço está incluído na célula.

::: não importante Apenas os serviços que foram encontrados em menos um dispositivo terá uma coluna mostrando seu estado.
:::

Botões

Os botões do modo de visualização, comuns a todas as seções, estão descritos na [Monitoramento](#) página.

Usando filtro

Você pode usar o filtro para exibir apenas as regras de descoberta que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

Sem nada selecionado no filtro, todas as regras de descoberta habilitadas são exibido. Para selecionar uma regra de descoberta específica para exibição, comece a digitar seu nome no filtro. Todas as regras de descoberta habilitadas correspondentes serão listados para seleção. Mais de uma regra de descoberta pode ser selecionada.

2 Services

Visão Geral¶

¶ O menu Serviços é usado para as funções de [service monitoring](#) do Zabbix.

1 Services

Visão geral



Nesta seção você pode ver o status consolidado de todos os serviços que foram configurado no Zabbix, baseado em sua infraestrutura.

Um serviço pode ser uma hierarquia que consiste em vários níveis de outros serviços, chamados serviços “filhos”, que são atributos ao status geral do service (veja também uma visão geral da funcionalidade de [service monitoring](#).)

As principais categorias de status do serviço são OK ou Problema, onde o status Problema é expresso pelo nome e cor da severidade do problema correspondente.

Enquanto o modo de visualização permite monitorar os serviços com seus status e outros detalhes, você também pode [configure](#) a hierarquia de serviço nesta seção (adicionar/editar serviços, serviços filho) alternando para o modo de edição.

Para passar do modo de visualização para o modo de edição (e voltar) clique no respectivo botão no canto superior direito:

-  - ver serviços
-  - adicionar/editar serviços e serviços filho

Observe que o acesso à edição depende de [user role](#) nas definições de permissão.

Viewing services

Services

View

Edit

Filter

Name	Status	Root cause	Created at	Tags
Availability 2	High	Nodata trigger, Nodata trigger 1h	2000-01-01	SLA: 3
Disc space	OK		2000-01-01	SLA: 1
Example service	OK		2000-01-01	SLA: 5

Displaying 3 of 3 found

A list of the existing services is displayed.

Displayed data:

Parameter	Description
Name	Service name. The service name is a link to service details . The number after the name indicates how many child services the service has.

Parameter	Description
Status	Service status: OK - no problems (trigger color and severity) - indicates a problem and its severity. If there are multiple problems, the color and severity of the problem with highest severity is displayed.
Root cause	Underlying problems that directly or indirectly affect the service status are listed. The same problems are listed as returned by the {SERVICE.ROOTCAUSE} macro. Click on the problem name to see more details about it in Monitoring → Problems. Problems that do not affect the service status are not in the list.
Created at	The time when the service was created is displayed.
Tags	Tags of the service are displayed. Tags are used to identify a service in service actions and SLAs.

Buttons




View mode buttons being common for all sections are described on the [Monitoring](#) page.

Using filter

You can use the filter to display only the services you are interested in.

Editando Serviços

Clique no botão Editar para acessar o modo de edição. Quando em modo de edição, a listagem é complementada com caixas de seleção antes de cada serviço e também com as seguintes opções adicionais:

-  - adicionar um serviço filho a este serviço
-  - editar este serviço
-  - excluir este serviço

Services

Create service View Edit

Filter

<input type="checkbox"/> Name	Status	Root cause	Created at	Tags	
<input type="checkbox"/> Availability 2	High	Nodata trigger, Nodata trigger 1h, Temperature is too high	2000-01-01	SLA: 3	+ ↗ x
<input type="checkbox"/> Disc space	OK		2000-01-01	SLA: 1	+ ↗ x
<input type="checkbox"/> Example service	OK		2000-01-01	SLA: 5	+ ↗ x

Displaying 3 of 3 found

Para [configure](#) um novo serviço, clique no botão Criar serviço no canto superior direito.

Service details

To access service details, click on the service name. To return to the list of all services, click on All services.

Service details include the info box and the list of child services.

All services / Availability

Info Filter

Availability

Parent services:
Status: High
SLA: SLA:3: 0
Tags: SLA: 3

	Reporting period	SLO	SLI	Uptime	Downtime	Error budget	Created at	Tags	
<input type="checkbox"/> Name	2022-01-09 – 01-15	100%	0	0	4d 10h 48m	-4d 10h 48m			
<input type="checkbox"/> Connections		High			Nodata trigger, Nodata trigger 1h		2000-01-01	SLA: 2 Type: Connection	+ ↗ x
<input type="checkbox"/> Servers		Warning			Temperature is too high		2000-01-01	SLA: 4 Type: CPU	+ ↗ x

Displaying 2 of 2 found

0 selected Mass update Delete




To access the info box, click on the Info tab. The info box contains the following entries:

- Names of parent services (if any)
- Current status of this service
- Current SLA(s) of this service, in the format SLA name:service level indicator. 'SLA name' is also a link to the SLA report for this service. If you position the mouse on the info box next to the service-level indicator (SLI), a pop-up info list is displayed with SLI details. The service-level indicator displays the current service level, in percentage.
- Service tags

The info box also contains a link to the [service configuration](#).

To use the filter for child services, click on the Filter tab.

When in edit mode, the child service listing is complemented with additional editing options:

-  - add a child service to this service
-  - edit this service
-  - delete this service

2 Ações de serviço

Visão geral

Na seção Serviços → Ações de serviço os usuários podem configurar e manter as ações de serviço [configure](#).

As ações configuradas serão exibidas na lista com respeito a [user role permissions](#). Os usuários somente poderão ver as ações para os serviços aos quais o papel do usuário concede acesso.

Os dados exibidos, filtros e as opções de edição em massa disponíveis são os mesmos que para outros tipos de ações [actions](#).

3 SLA

Visão geral

Esta seção permite visualizar e [configure](#) SLAs.

Uma lista dos SLAs configurados é exibida. Observe que apenas os SLAs relacionados a serviços acessíveis ao usuário serão exibidos (como somente leitura, a menos que Gerenciar SLA esteja ativado nas funções de permissão de usuário).

Dados exibidos:

Parâmetro	Descrição
Nome	O nome do SLA é exibido. O nome é um link para configuração do SLA .
SLO	O objetivo de nível de serviço (SLO) é exibido.
Data efetiva	A data de início do cálculo do SLA é exibida.
Período do relatório	O período usado no relatório de SLA é exibido - diário, semanal, mensal, trimestral ou anual.
Fuso horário	O fuso horário do SLA é exibido.
Programação	A programação do SLA é exibida - 24x7 ou personalizada.
Relatório de SLA	Clique no link para ver o relatório de SLA para este SLA.
Status	O status do SLA é exibido - ativado ou desativado.

4 Relatório de SLA

Visão geral

Esta seção permite visualizar relatórios de SLA, com base nos critérios selecionados no filtro.

Os relatórios de SLA também podem ser exibidos como um [dashboard widget](#).

Relatório

O filtro permite selecionar o relatório com base no nome do SLA, bem como no Nome do Serviço. Também é possível limitar o período exibido.

SLA report

Filter

SLA

SLA:3

×

Select

From

YYYY-MM-DD

⋮

Service

type here to search

Select

To

YYYY-MM-DD

⋮

Apply

Reset

		2020-06	2020-07	2020-08	2020-09	2020-10	2020-11	2020-12	2021-01	2021-02	2021-03	2021-04	2021-05	2021-06	2021-07	2021-08	2021-09	2021-10	2021-11	2021-12	2022-01	
Service	SLO	100%	100	100	100	100	100	100	100	100	100	100	100	100	100	72.5434	0.0028	28.8072	17.049	0	0	0
Availability		100%	100	100	100	100	100	100	100	100	100	100	100	100	100	72.5434	0.0028	28.8072	17.049	0	0	0

Displaying 1 of 1 found

Cada coluna (período) exibe o SLI para esse período. SLIs que estão em violação do conjunto SLO são destacados em vermelho. 20 períodos são exibidos no relatório. Um máximo de 100 períodos podem ser exibidos, se tanto as opções de data De e Até forem especificadas.

Detalhes do relatório

Se você clicar no nome do serviço no relatório, poderá acessar outro relatório que exibe uma visão mais detalhada.

SLA report

Filter						
SLA		SLA:3	Select		From	
Service		Availability	Select		To	
		Apply		Reset		
Month	SLO	SLI	Uptime	Downtime	Error budget	Excluded downtimes
2022-01	100%	0	0	12d 16h 16m	-12d 16h 16m	
2021-12	100%	0	0	1m 1d	-1m 1d	
2021-11	100%	0	0	1m	-1m	
2021-10	100%	17.049	5d 6h 50m	25d 17h 9m	-25d 17h 9m	
2021-09	100%	28.8072	8d 15h 24m	21d 8h 35m	-21d 8h 35m	
2021-08	100%	0.0028	1m 15s	1m 23h	-1m 23h	
2021-07	100%	72.5434	22d 11h 43m	8d 12h 16m	-8d 12h 16m	
2021-06	100%	100	1m	0	0	
2021-05	100%	100	1m 1d	0	0	
2021-04	100%	100	1m	0	0	
2021-03	100%	100	1m 1d	0	0	
2021-02	100%	100	28d	0	0	

3 Inventário

Visão geral

O menu inventário fornece uma visão geral de inventário dos hosts com possibilidade de ver visão resumida ou detalhada.

1 Visão geral

Através do módulo Inventário -> Visão geral você pode visualizar uma visão geral do **inventário de hosts**.

Para que a visão geral seja apresentada, selecione um grupo de host (ou todos) e o campo do inventário que irá agrupar os dados. Será apresentada a quantidade de hosts correspondentes a valor localizado naquele campo do inventário.

Host inventory overview

Filter	
Host groups	type here to search
Grouping by	Type
Apply Reset	
Type	Host count
Server	4
Zabbix server	1

A completude de uma visão geral depende de quanta informação estiver disponível no inventário dos hosts.

Os números na coluna Contagem de hosts possuem links; eles levam à um relatório detalhado com os hosts que foram contabilizados na visão geral.

Host inventory

Filter

Host groups

type here to search

Select

Field

Type

contains

Zab

Apply

Reset

Host	Group	Name	Type	OS	Serial number A	Tag	MAC address A
Zabbix server	Zabbix servers	martins-hp	Zabbix server	Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP			

Displaying 1 of 1 found

Visão geral

A seção Inventário → Visão geral fornece maneiras de ter uma visão geral de dados de **inventário de host**.

Para que uma visão geral seja exibida, selecione grupos de hosts (ou nenhum) e o campo de inventário pelo qual exibir dados. O número de hospedeiros correspondente a cada entrada do campo escolhido será exibido.

Host inventory overview

Filter

Host groups

type here to search

Select

Grouping by

Type

Apply

Reset

Type	Host count
Server	4
Zabbix server	1

A integridade de uma visão geral depende de quanto estoque informações são mantidas com os anfitriões.

Os números na coluna Host count são links; eles levam a esses hospedeiros sendo filtrado na tabela Host Inventories.

Host inventory

Filter

Host groups

type here to search

Select

Field

Type

equals

Server

Apply

Reset

Host	Group	Name	Type	OS	Serial number A	Tag	MAC address A
Zabbix server	Zabbix servers	martins-hp	Zabbix server	Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP			

Displaying 1 of 1 found

2 Hosts

Visão geral

Na seção Inventário → Hosts **inventário data** dos hosts são exibidos.

Você pode filtrar os hosts por grupo(s) de hosts e por qualquer campo de inventário para exibir apenas os hosts nos quais você está interessado.

Host inventory

Filter

Host groups

Field contains

Host	Group	Name	Type	OS	Serial number	Tag	MAC address
Zabbix server	Zabbix servers	martins-hp	Zabbix server	Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP			

Displaying 1 of 1 found

Para exibir todos os inventários de host, selecione nenhum grupo de host no filtro, limpe o campo de comparação no filtro e pressione "Filtro".

Embora apenas alguns campos-chave do inventário sejam exibidos na tabela, você pode também visualizar todas as informações de inventário disponíveis para esse host. Fazer isso, clique no nome do host na primeira coluna.

Detalhes do inventário

A guia **Visão geral** contém algumas informações gerais sobre o host juntamente com links para scripts predefinidos, dados de monitoramento mais recentes e opções de configuração:

Host inventory

Overview

Details

Host name

Zabbix server

Agent interfaces

IP address	DNS name	Connect to	Port
127.0.0.1		<input type="button" value="IP"/> <input type="button" value="DNS"/>	10050

SNMP interfaces

127.0.0.1		<input type="button" value="IP"/> <input type="button" value="DNS"/>	161
-----------	--	--	-----

OS

Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP

Monitoring

[Web](#) [Latest data](#) [Problems](#) [Graphs](#) [Dashboards](#)

Configuration

Host [Items 148](#) [Triggers 67](#) [Graphs 28](#) [Discovery 4](#) [Web 1](#)

A guia **Detalhes** contém todos os detalhes de inventário disponíveis para o hospedeiro:

Overview

Details

Type

Zabbix server

Name

martins-hp

OS

Linux version 5.3.0-46-generic (buildd@lcy01-amd64-013) (gcc version 7.5.0 (Ubuntu 7.5.0-3ubuntu1~18.04)) #38~18.04.1-Ubuntu SMP

A integridade dos dados de inventário depende de quanto inventário informações são mantidas com o host. Se nenhuma informação for mantida, a guia Detalhes é desabilitada.

4 Relatórios

Visão geral

O menu Relatórios apresenta várias seções que contêm uma variedade de relatórios predefinidos e personalizáveis pelo usuário, focados em exibir uma visão geral de parâmetros como informações do sistema, gatilhos (triggers) e dados coletados.

1 Informações do sistema

Visão geral

Em Relatórios → Informações do sistema um resumo do servidor Zabbix chave e dados do sistema são exibidos.

Observe que em uma configuração de alta disponibilidade, é possível redirecionar o fonte de informações do sistema (instância do servidor) editando a arquivo ui/conf/zabbix.conf.php - descomente e defina \$ZBX_SERVER, \$ZBX_SERVER_PORT para um servidor diferente do mostrado ativo.

Com a configuração de alta disponibilidade habilitada, um bloco separado é exibido abaixo das estatísticas do sistema com detalhes de nós de alta disponibilidade. este bloco é visível apenas para usuários do Zabbix Super Admin.

Informações do sistema também estão disponíveis como um painel **widget**.

Estatísticas do sistema

System information

Parameter		Value	Details
Zabbix server is running		Yes	192.168.8.103:10051
Number of hosts (enabled/disabled)		5	4 / 1
Number of templates		140	
Number of items (enabled/disabled/not supported)		199	155 / 29 / 15
Number of triggers (enabled/disabled [problem/ok])		89	87 / 2 [8 / 79]
Number of users (online)		3	1
Required server performance, new values per second		1.96	
High availability cluster		Enabled	Fail-over delay: 1 minute
Name	Address	Last access	Status
base	192.168.8.103:10051	2s	Active
base2	localhost:10051	5m 11s	Stopped

Dados exibidos:

Parâmetro	Valor	Detalhes
O servidor Zabbix está rodando	Status do servidor Zabbix: Sim - o servidor está rodando Não - o servidor não está rodando Nota: Para exibir o resto as informações que o web frontend precisa que o servidor esteja executando e deve haver pelo menos um processo trapper iniciado no servidor (parâmetro StartTrappers no arquivo zabbix_server.conf > 0).	Localização e porta do servidor Zabbix.
Número de hosts	O número total de hosts configurados é exibido.	Número de hosts monitorados/hosts não monitorados.
Número de modelos	O número total de modelos é exibido.	
Número de itens	O número total de itens é exibido.	Número de itens monitorados/desativados/sem suporte. Os itens em hosts desativados são contados como desativados.

Parâmetro	Valor	Detalhes
Número de acionadores	O número total de acionadores é exibido.	Número de acionadores ativados/desativados. [Triggers in problem/ok state.] Triggers atribuídos a hosts desabilitados ou dependendo de itens desabilitados são contados como desabilitados.
Número de usuários	É exibido o número total de usuários configurados.	Número de usuários online.
Desempenho do servidor necessário, novos valores por segundo	É exibido o número esperado de novos valores processados pelo servidor Zabbix por segundo.	Desempenho do servidor necessário é uma estimativa e pode ser útil como diretriz. Para números precisos de valores processados, use o <code>zabbix[wcache,values,all]</code> internal item . Itens ativados de hosts monitorados estão incluídos em o cálculo. Os itens de log são contados como um valor por intervalo de atualização do item. Os valores de intervalos regulares são contados; os valores de intervalo flexível e de agendamento não são. O cálculo não é ajustado durante um período de manutenção "nodata". Itens de caçador não são contados.
Tabelas de histórico do banco de dados atualizadas	Status de atualização do banco de dados: Não - as tabelas de histórico do banco de dados não foram atualizadas	Este campo é exibido se a atualização do banco de dados para o intervalo estendido para valores numéricos (flutuantes) não tiver sido concluída. Consulte instruções para habilitar um intervalo estendido de valores numéricos (float).
Cluster de alta disponibilidade	Status de cluster de alta disponibilidade para o servidor Zabbix: desativado - servidor autônomo ativado - pelo menos existe um nó de alta disponibilidade	Se ativado, o atraso de failover é exibido.

Informações do sistema também exibirão uma mensagem de erro se:

- o banco de dados usado não possui o conjunto de caracteres necessário ou agrupamento (UTF-8);
- a versão do banco de dados está abaixo ou acima **suportado range** (disponível apenas para usuários com o tipo de função Superadministrador).

Nós de alta disponibilidade

Se **cluster de alta disponibilidade** estiver ativado, então outro bloco de dados é exibido com o status de cada alta nó de disponibilidade.

Name	Address	Last access	Status
node-active	192.168.1.13:10051	12s	Active
node6	192.168.1.10:10053	1h 2m 40s	Unavailable
node7	192.168.1.11:10053	3m 40s	Unavailable
node4	192.168.1.8:10052	1h 34m 29s	Stopped
node5	192.168.1.9:10053	1h 9m 51s	Stopped
node8	192.168.1.12:10051	21m 16s	Stopped
node1	192.168.1.5:10051	17s	Standby
node2	192.168.1.6:10051	16s	Standby
node3	192.168.1.7:10052	16s 2021-10-20 17:58:47	Standby

Dados exibidos:

Coluna	Descrição
Nome	Nome do nó, conforme definido na configuração do servidor.
Endereço	Endereço IP e porta do nó.
Último acesso	Hora do último acesso do nó. Passar o mouse sobre a célula mostra o carimbo de data/hora do último acesso em formato longo.
Status	Status do nó: Ativo - o nó está ativo e funcionando Indisponível - o nó não foi visto por mais de um atraso de failover (você pode querer descobrir por que) Parado - o nó foi interrompido ou não pôde ser iniciado (você pode iniciá-lo ou excluí-lo) Em espera - o nó está ativo e aguardando

2 Relatórios agendados

Visão geral

Em Relatórios → Relatórios agendados usuários com suficiente podem configurar geração programada de versões em PDF dos dashboards, que serão enviado por e-mail para destinatários especificados.

Scheduled reports

Create report

Filter

Name

Show

All

Created by me

Status

Any

Enabled

Disabled

Expired

Apply

Reset

<input type="checkbox"/> Name ▲	Owner	Repeats	Period	Last sent	Status	Info
<input type="checkbox"/> Global view daily	Admin (Zabbix Administrator)	Daily	Previous day	Never	Enabled	

Displaying 1 of 1 found

A tela de abertura exibe informações sobre relatórios programados, que pode ser filtrado para facilitar a navegação - veja **Usando filter** abaixo.

Dados exibidos:

Coluna	Descrição
Nome	Nome do relatório
Proprietário	Usuário que criou o relatório
Repete	Frequência de geração do relatório (diário/semanal/mensal/anual)
Período	Período para o qual o relatório é elaborado
Último envio	A data e hora em que o último relatório foi enviado
Status	Status atual do relatório (ativado/desativado/expirado). Usuários com permissões suficientes podem alterar o status clicando nele - de Ativado para Desativado (e vice-versa); de Expirado para Desativado (e de volta). Exibido como um texto para usuários com direitos insuficientes.

Coluna	Descrição
Info	Exibe ícones informativos: Um ícone vermelho indica que a geração do relatório falhou; passar o mouse sobre ele exibirá uma dica de ferramenta com as informações do erro. Um ícone amarelo indica que um relatório foi gerado, mas o envio para alguns (ou todos) destinatários falhou ou que um relatório expirou; passar o mouse sobre ele exibirá uma dica de ferramenta com informações adicionais

Usando filtro

Você pode usar o filtro para restringir a lista de relatórios. Para melhor desempenho da pesquisa, os dados são pesquisados com macros não resolvidas.

As seguintes opções de filtragem estão disponíveis:

- Nome - a correspondência parcial de nomes é permitida;
- Proprietário do relatório - criado pelo usuário atual ou todos os relatórios;
- Status - selecione entre qualquer (mostrar todos os relatórios), ativado, desativado, ou expirado.

O filtro está localizado acima da barra de relatórios programados. Ele pode ser aberto e recolhido clicando na guia Filtro no canto superior direito.

Atualização em massa

Às vezes, você pode querer alterar o status ou excluir vários relatórios em uma vez. Em vez de abrir cada relatório individual para edição, você pode usar a função de atualização em massa para isso.

Para atualizar alguns relatórios em massa, faça o seguinte:

- Marque as caixas de seleção dos relatórios a serem atualizados na lista
- Clique no botão necessário abaixo da lista para fazer alterações (Ativar, Desabilitar ou Excluir).

3 Relatório de disponibilidade

Visão geral

Em Relatórios → Relatório de disponibilidade você pode ver qual proporção de tempo cada gatilho esteve em estado de problema/ok. A porcentagem de tempo para cada estado é exibido.

Assim, é fácil determinar a situação de disponibilidade de vários elementos do seu sistema.

Availability report

Mode By host

Host groups

type here to search

Select

Hosts

type here to search

Select

Apply

Reset

Host	Name	Problems	Ok	Graph
Zabbix server	/: Disk space is critically low (used > 90%)		100.0000%	Show
Zabbix server	/: Disk space is low (used > 80%)	0.0556%	99.9444%	Show
Zabbix server	/: Running out of free inodes (free < 10%)		100.0000%	Show
Zabbix server	/: Running out of free inodes (free < 20%)		100.0000%	Show
Zabbix server	/etc/passwd has been changed		100.0000%	Show
Zabbix server	Configured max number of open filedescriptors is too low (< 256)		100.0000%	Show

No menu suspenso no canto superior direito, você pode escolher o modo de seleção - seja para exibir gatilhos por hosts ou por gatilhos pertencente a um modelo.

Availability report

Mode By trigger template

Template group

all

Template

all

Template trigger

all

Host group

all

Apply

Reset

Host

Name

Problems

Ok

Graph

My host

/etc/passwd has been changed

100.0000%

Show

My host

Configured max number of open filedescriptors is too low (< 256)

100.0000%

Show

My host

Configured max number of processes is too low (< 1024)

100.0000%

Show

My host

Getting closer to process limit (over 80% used)

100.0000%

Show

My host

High CPU utilization (over 90% for 5m)

100.0000%

Show

My host

High memory utilization (>90% for 5m)

100.0000%

Show

My host

High swap space usage (less than 50% free)

100.0000%

Show

My host

Lack of available memory (< 20M of 15.54 GB)

100.0000%

Show

My host

Load average is too high (per CPU load over 1.5 for 5m)

100.0000%

Show

O nome do gatilho é um link para os eventos mais recentes desse gatilho.

Usando filtro

O filtro pode ajudar a diminuir o número de hosts e/ou gatilhos exibido. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O filtro está localizado abaixo da barra Relatório de disponibilidade. Pode ser aberto e recolhido clicando na guia Filtro à esquerda.

Filtrando por modelo de gatilho

No modo por modelo de acionador, os resultados podem ser filtrados por um ou vários parâmetros listados abaixo.

Parâmetro	Descrição
Grupo de modelos	Selecione todos os hosts com gatilhos de modelos pertencentes a esse grupo. Qualquer grupo de hosts que inclua pelo menos um modelo pode ser selecionado.
Modelo	Selecione hosts com gatilhos do modelo escolhido e todos os modelos aninhados. Apenas os acionadores herdados do modelo selecionado serão exibidos. Se um modelo aninhado tiver gatilhos próprios adicionais, esses gatilhos não serão exibidos.
//Template trigger //	Selecione hosts com o trigger escolhido. Outros gatilhos dos hosts selecionados não serão exibidos.
Grupo de hosts	Selecione os hosts pertencentes ao grupo.

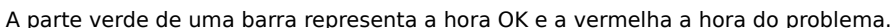
Filtrando por host

No modo by host os resultados podem ser filtrados por um host ou pelo host grupo. Especificar um grupo de hosts pai seleciona implicitamente todos os hosts aninhados grupos.

Seletor de período de tempo

O período de tempo seletor]([manual/config/visualization/graphs/simple#time_period_selector) permite selecionar períodos frequentemente necessários com um clique do mouse. A Hora seletor de período pode ser aberto clicando na guia de período de tempo ao lado de o filtro.

Clicar em Mostrar na coluna Gráfico exibe um gráfico de barras onde informações de disponibilidade são exibidas em formato de barra cada barra representando uma semana passada do ano atual.



Visão geral

≡ 100 busiest triggers

As entradas de coluna de host e gatilho são links que oferecem algumas opções:

- ## Usando filtro

O filtro está localizado abaixo da barra 100 triggers mais ocupados. Pode ser aberto e recolhido clicando na guia Filtro à esquerda.

O período de tempo seletor[/manual/config/visualization/graphs/simple#time_period_selector) permite selecionar períodos frequentemente necessários com um clique do mouse. A Hora seletor de período pode ser aberto clicando na guia de período de tempo ao lado de o filtro.

5 Auditoria

Na seção Relatórios → Auditoria os usuários podem visualizar os registros das alterações feitas no front-end.

Note:

O log de auditoria deve ser habilitado na Administração **configurações** para exibir registros de auditoria. Se o registro estiver desabilitado, o histórico do frontend alterações não são registradas no banco de dados e os registros de auditoria não podem ser visto.

Audit log

Time	User	IP	Resource	ID	Action	Recordset ID	Details
2022-05-30 12:07:34	Admin	127.0.0.1	User	4	Update	cl3sicbqq0000z8ep87xz41zs	Description: Database manager user.lang: default => en_GB
2022-05-30 12:07:13	Admin	127.0.0.1	User	1	Login	cl3sibvqn0000z8ep40q8w1k	
2022-05-30 12:07:13	guest	127.0.0.1	User	2	Failed login	cl3sibvqn0000z8ep40q8w1k	
2022-05-30 12:07:12	guest	127.0.0.1	User	2	Failed login	cl3sibvem0000z8epv1m1xizi	

O log de auditoria exibe os seguintes dados:

Coluna	Descrição
Hora	Carimbo de data e hora do registro de auditoria.
Usuário	Usuário que realizou a atividade.
IP	IP a partir do qual a atividade foi iniciada.
Recurso	Tipo do recurso afetado (host, grupo de hosts, etc.).
Ação	Tipo de atividade: Login, Logout, Adicionado, Atualizado, Excluído, Ativado ou Desativado.
ID	ID do recurso afetado. Clicar no hiperlink resultará na filtragem de registros de log de auditoria por essa ID de recurso.
ID do conjunto de registros	ID compartilhado para todos os registros de log de auditoria criados como resultado da mesma operação de front-end. Por exemplo, ao vincular um modelo a um host, um registro de log de auditoria separado é criado para cada entidade de modelo herdada (item, gatilho etc.) - todos esses registros terão o mesmo ID do conjunto de registros. Clicar no hiperlink resultar na filtragem de registros de log de auditoria por este ID do conjunto de registros.
Detalhes	Descrição do recurso e informações detalhadas sobre a atividade realizada. Se um registro contiver mais de duas linhas, um link adicional Detalhes será exibido. Clique neste link para ver a lista completa de alterações.

Usando filtro

Você pode usar o filtro para restringir os registros por usuário, atividade tipo, recurso afetado e operação de front-end (consulte ID do conjunto de registros). Para melhor desempenho de pesquisa, todos os dados são pesquisados com macros não resolvidas.

O filtro está localizado abaixo da barra Registro de auditoria. Pode ser aberto e recolhido clicando na guia Filtro no canto superior direito.

Seletor de período

O período seletor[/manual/config/visualization/graphs/simple#time_period_selector) permite selecionar períodos frequentemente necessários com um clique do mouse. A Hora seletor de período pode ser aberto clicando na guia de período ao lado de o filtro.

Visão Geral

Na Seção Relatórios → Auditoria os usuários podem visualizar os registros das alterações feitas no front-end.

Note:

O log de auditoria deve ser habilitado na Administração **Configurações** para exibir registros de auditoria. Se o registro estiver desabilitado, o histórico de alterações do frontend não são registradas no banco de dados e os registros de auditoria não podem ser vistos.

Audit log

Time	User	IP	Resource	ID	Action	Recordset ID	Details
2022-05-30 12:07:34	Admin	127.0.0.1	User	4	Update	ci3sicbqq0000z8ep87xz41zs	Description: Database manager user.lang: default => en_GB
2022-05-30 12:07:13	Admin	127.0.0.1	User	1	Login	ci3sibvqn0000z8ep40q8w1k	
2022-05-30 12:07:13	guest	127.0.0.1	User	2	Failed login	ci3sibvqn0000z8ep40q8w1k	
2022-05-30 12:07:12	guest	127.0.0.1	User	2	Failed login	ci3sibvem0000z8epvm1xizi	

O log de auditoria exibe os seguintes dados::

Column	Description
Hora	Data e Hora do registro de auditoria.
Usuário	Usuário que realizou a atividade.
IP	IP a partir do qual a atividade foi iniciada.
Recurso	Tipo do recurso afetado (host, grupo de hosts, etc.).
Ação	Tipo de atividade: Login, Logout, Adicionado, Atualizado, Excluído, Ativado ou Desativado.
ID	ID do recurso afetado. Clicar no link resultará em registros do log de auditoria filtrados pelo ID de recurso.
Conjunto de ID's	ID compartilhado para todos os registros de log de auditoria criados como resultado da mesma operação de front-end. Por exemplo, ao vincular um modelo a um host, um registro de log de auditoria separado é criado para cada entidade de modelo herdada (item, gatilho etc.) - todos esses registros terão o mesmo ID do conjunto de registros. Clicar no link resultará na filtragem de registros de log de auditoria por este ID do conjunto de registros.
Detalhes	Descrição do recurso e informações detalhadas sobre a atividade realizada. Se um registro contiver mais de duas linhas, um link adicional Detalhes será exibido. Clique neste link para ver a lista completa de alterações.

Usando Filtro

O filtro está localizado abaixo da barra Registro de auditoria. Pode ser aberto e recolhido clicando na guia Filtro no canto superior direito.

Zoom out

Last 3 hours

Filter

Users

Select

Resource

All

Resource ID

Recordset ID

Actions

☐ Add

☒ Failed login

☐ Logout

☐ Delete

☐ History clear

☐ Update

☐ Execute

☒ Login

Apply

Reset

Você pode usar o filtro para restringir os registros por usuário, recurso afetado, ID do recurso e operação de front-end (ID do conjunto de registros). Você também pode selecionar a ação (por exemplo, adicionar, atualizar, excluir etc.) para o recurso. Desde o Zabbix 6.0.5 uma ou mais ações podem ser selecionadas.

Para um melhor desempenho de pesquisa, todos os dados são pesquisados com macros não resolvidas.

Seletor de período

O **Seletor de Período** permite selecionar períodos usados frequentemente com um clique do mouse. A Hora seletor de período pode ser aberto clicando na guia de período ao lado de o filtro.

6 Log de ações

Visão geral

Na seção Relatórios → Log de ações, os usuários podem visualizar detalhes das operações (notificações, comandos remotos) executados dentro de uma ação.

Action log

<div> <div>Recipients <input type="text"/></div> <div>Select</div> </div> <div> <div>Apply</div> <div>Reset</div> </div>					
Time	Action	Type	Recipient	Message	Status Info
2020-06-09 15:47:16	Report problems to Zabbix administrators	Email	Admin (Zabbix Administrator) <i>marina.generalova@zabbix.com</i>	Subject: Resolved in 2m: High CPU utilization (over 75% for 5m) Message: Problem has been resolved at 15:47:13 on 2020.06.09 Problem name: High CPU utilization (over 75% for 5m) Problem duration: 2m Host: Zabbix server Severity: Warning Original problem ID: 1287	Sent
2020-06-09 15:44:40	Report problems to Zabbix administrators	Email	Admin (Zabbix Administrator) <i>marina.generalova@zabbix.com</i>	Subject: Resolved in 3m: Zabbix agent is not available (for 1m) Message: Problem has been resolved at 15:44:37 on 2020.06.09 Problem name: Zabbix agent is not available (for 1m) Problem duration: 3m Host: Zabbix server Severity: Average Original problem ID: 1286	Sent

Dados exibidos:

Coluna	Descrição
Hora	Carimbo de data e hora da operação.
Ação	O nome da ação que causa as operações é exibido.
Tipo	O tipo de operação é exibido - Email ou Command.
Destinatário(s)	Nome de usuário, nome, sobrenome (entre parênteses) e endereço de e-mail do destinatário da notificação são exibidos.
Mensagem	O conteúdo da mensagem/comando remoto é exibido. Um comando remoto é separado do host de destino com um símbolo de dois pontos: <host>: <comando>. Se o comando remoto for executado no servidor Zabbix, as informações terão o seguinte formato: Zabbix server: <command>
Status	O status da operação é exibido: Em andamento - a ação está em andamento Para ações em andamento, o número de tentativas restantes é exibido - o número restante de vezes que o servidor tentará enviar a notificação . Enviado - a notificação foi enviada Executado - o comando foi executado Não enviado - a ação não foi concluída.
Info	Informações de erro (se houver) sobre a execução da ação são exibidas.

Usando filtro

Você pode usar o filtro para restringir os registros pela mensagem destinatário(s). Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O filtro está localizado abaixo da barra Registro de ações. Pode ser aberto e recolhido clicando na guia Filtro à esquerda.

Seletor de período

O período de tempo seletor[/manual/config/visualization/graphs/simple#time_period_selector) permite selecionar períodos frequentemente necessários com um clique do mouse. A Hora seletor de período pode ser aberto clicando na guia de período de tempo ao lado de o filtro.

7 Notificações

Visão geral

Na seção Relatórios → Notificações um relatório sobre o número de notificações enviadas a cada usuário são exibidas.

Nas listas suspensas no canto superior direito, você pode escolher a mídia tipo (ou todos), período (dados para cada dia/semana/mês/ano) e ano para as notificações enviadas.

Notifications					Media type	all	Period	Monthly	Year	2020
Month					Admin (Zabbix Administrator)	Database manager	guest	user (New User)		
January										
February										
March										
April					48					
May					568					

Cada coluna exibe os totais por um usuário do sistema.

5 Configuração

Visão geral

O menu Configuração contém seções para configurar o Zabbix principal funções, como hosts e grupos de hosts, coleta de dados, limites, envio de notificações de problemas, criação de visualização de dados e outros.

1 Itens

Visão geral

A lista de itens para um template pode ser acessada através de Configuration → Templates, clicando nos Itens para o template correspondente.

Uma lista de itens existente será exibida.

Items

Create item

All templates / Template OS Linux by Zabbix agen...

Items 41

Triggers 14

Graphs 8

Dashboards 1

Discovery rules 3

Web scenarios

Filter

<input type="checkbox"/>	Name	Triggers	Key ▲	Interval	History	Trends	Type	Status	Tags
<input type="checkbox"/>	... Template Module Zabbix agent active: Host name of Zabbix agent running		agent.hostname	1h	7d		Zabbix agent (active)	Enabled	Application: Monitorin...
<input type="checkbox"/>	... Template Module Zabbix agent active: Zabbix agent ping	Triggers 1	agent.ping	1m	7d	365d	Zabbix agent (active)	Enabled	Application: Status
<input type="checkbox"/>	... Template Module Zabbix agent active: Version of Zabbix agent running		agent.version	1h	7d		Zabbix agent (active)	Enabled	Application: Monitorin...
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent active: Maximum number of open file descriptors	Triggers 1	kernel.maxfiles	1h	7d	365d	Zabbix agent (active)	Enabled	Application: General
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent active: Maximum number of processes	Triggers 2	kernel.maxproc	1h	7d	365d	Zabbix agent (active)	Enabled	Application: General
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent active: Number of processes	Triggers 1	proc.num	1m	7d	365d	Zabbix agent (active)	Enabled	Application: General
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent active: Number of running processes		proc.num[,run]	1m	7d	365d	Zabbix agent (active)	Enabled	Application: General

Dados exibidos:

Column	Descrição
Item menu	<p>Clique no ícone three-dot para abrir o menu desse item específico com essas opções:</p> <p>Criar trigger - criar um trigger com base nesse item</p> <p>Triggers - clique para ver uma lista com links para um trigger já configurado deste item</p> <p>Criar item dependente - criar um item dependente para esse item</p> <p>Criar regra de descoberta dependente - criar uma regra de descoberta dependente para esse item</p>
Template	<p>Template ao qual o item pertence.</p> <p>Esta coluna será exibida somente se vários templates forem selecionados no filtro.</p>
Name	<p>Nome do item exibido como um link azul para os detalhes do item.</p> <p>Clicando no link, você abre a configuração do item configuration form.</p> <p>Se o item for herdado de um outro template, o nome do template será exibido em um link cinza, antes do nome do item. Clicando no link do template você abrirá a lista de item no nível daquele template.</p>
Triggers	<p>Mover o mouse sobre os Triggers exibirá uma caixa de informação mostrando os triggers associados ao item.</p> <p>O número de triggers será exibido em cinza.</p>

Column	Descrição
Key	A chave do item é exibida.
Interval	Frequência de verificação é exibida.
History	Quantos dias o histórico de dados do item mantido é exibido.
Trends	Quantos dias o histórico de trends mantido é exibido.
Type	Tipo de item é exibido (Zabbix agent, SNMP agent, simple check, etc).
Status	Status do item é exibido - Habilitado ou Desabilitado. Ao clicar no status, você consegue alterá-lo - de Habilitado para Desabilitado (e vice-versa).
Tags	As tags do item são exibidas. Até três tags (nome: valores pairs) podem ser exibidas. Se haver mais tags, um "..." link que permite ver todas as tags será exibido.

Para configurar um item novo, clique no botão Criar item no canto superior a direita.

Opções de edição em massa

Os botões abaixo oferecem algumas opções de edição em massa:

- Habilitar - alterar o status do item para Habilitado.
- Desabilitar - alterar o status do item para Desabilitado.
- Copiar - copiar os itens para outros hosts e templates.
- Atualizar em massa - **update several properties** para vários itens de uma vez.
- Excluir - excluir os itens.

Para utilizar essas opções, marque as caixas de seleção antes dos respectivos itens e, em seguida, clique no botão necessário.

Usando filtro

A lista de itens pode conter muitos itens. Ao utilizar o filtro, você consegue filtrar alguns deles para localizar rapidamente os itens que está procurando.

Para melhorar o desempenho da pesquisa, os dados são pesquisados com macros não resolvidos.

O ícone Filtro está disponível no canto superior a direita. Ao clicar nesse ícone, você abrirá um filtro onde você pode especificar os critérios de filtragem desejados.

Parameter	Descrição
Host groups	Filtre um ou mais grupos de host. Apenas grupos de host que contêm pelo menos um template podem ser selecionados. Especificar um grupo de host parent seleciona implicitamente todos grupos de host aninhados.
Templates	Filtre por um ou mais templates.
Name	Filtre pelo nome do item.
Key	Filtre por chave do item.

Parameter	Descrição
Value mapping	Filtre pelo mapa de valores utilizado. Este parâmetro não será exibido se a opção de Templates estiver vazia.
Type	Filtre pelo tipo do item (Zabbix agent, SNMP agent, etc.).
Type of information	Filtre pelo tipo da informação (Numeric unsigned, float, etc.).
History	Filtre pelo tempo do histórico de itens.
Trends	Filtrar pelo tempo das trends de itens.
Update interval	Filtre pelo intervalo de atualização do item.
Tags	Especifique as tags para limitar o números de itens exibidos. É possível incluir e excluir tags específicas e valores de tags. Muitas condições podem ser configuradas. O nome da Tag correspondente é sempre maiúscula-minúscula. Há vários operadores disponíveis para cada condição. Existe - incluir nomes das tags específicas Igual - incluir nomes das tags específicas e valores de tags (maiúscula-minúscula) Contém - incluir nomes de tags específicas onde os valores de tags contêm a string inserida (correspondência de substring, case-insensitive) Não existe - excluir nomes das tags específicas Não é igual - excluir tags específicas e valores de tags (case-sensitive) Não contém - excluir nomes de tags específicas onde os valores contêm uma string inserida (correspondência de substring, case-insensitive) Há dois tipos de cálculos para condições: E/Ou - todas as condições devem ser atendidas, condições que contêm o mesmo nome da tag serão agrupadas pela condição Ou - se uma condição for atendida é suficiente
Status	Filtre pelo status do item - Habilitado ou Desabilitado.
Triggers	Filtre itens com (ou sem) triggers.
Inherited	Filtre itens herdados (ou não herdados) de templates correspondentes.

O **Subfilter** abaixo do filtro oferece opções adicionais de filtragem (para dados já filtrados). Você pode selecionar grupos de itens com um valor de parâmetro comum. Ao clicar em um grupo, ele é destacado e apenas os itens com esse valor de parâmetro permanecem na lista.

2 Triggers

Visão geral

A lista de gatilhos para um modelo pode ser acessada em Configuração → Modelos clicando em Triggers para o respectivo modelo.

Triggers						Create trigger
All templates / Linux OS agent Items 42 Triggers 14 Graphs 8 Dashboards 1 Discovery rules 3 Web scenarios						Filter
Severity	Name	Operational data	Expression	Status	Tags	
Information	Template Module Linux generic by Zabbix agent: /etc/passwd has been changed Depends on: Linux OS agent: Operating system description has changed Linux OS agent: System name has changed (new name: {ITEM.VALUE})		<code>(last(/Linux OS agent/vfs.file.cksum[/etc/passwd],#1)<=last(/Linux OS agent/vfs.file.cksum[/etc/passwd],#2))>0</code>	Enabled		
Information	Template Module Linux generic by Zabbix agent: Configured maximum number of open file descriptors is too low (< {\${KERNEL.MAXFILES.MIN}})		<code>last(/Linux OS agent/kernel.maxfiles)<{\${KERNEL.MAXFILES.MIN}}</code>	Enabled		
Information	Template Module Linux generic by Zabbix agent: Configured maximum number of processes is too low (< {\${KERNEL.MAXPROC.MIN}}) Depends on: Linux OS agent: Getting closer to process limit (over 80% used)		<code>last(/Linux OS agent/kernel.maxproc)<{\${KERNEL.MAXPROC.MIN}}</code>	Enabled		
Warning	Template Module Linux generic by Zabbix agent: Getting closer to process limit (over 80% used)	{ITEM.LASTVALUE1} active, {ITEM.LASTVALUE2} limit.	<code>last(/Linux OS agent/proc.num)/last(/Linux OS agent/kernel.maxproc)*100>80</code>	Enabled		
Warning	Template Module Linux CPU by Zabbix agent: High CPU utilization (over {\${CPU.UTIL.CRIT}}% for 5m) Depends on: Linux OS agent: Load average is too high (per CPU load over {\${LOAD_AVG_PER_CPU.MAX.WARN}} for 5m)	Current utilization: {ITEM.LASTVALUE1}	<code>min(/Linux OS agent/system.cpu.util,5m)>{\${CPU.UTIL.CRIT}}</code>	Enabled		

Dados exibidos:

Coluna	Descrição
Gravidade	A gravidade do acionador é exibida pelo nome e pela cor de fundo da célula.
Modelo	Modelo ao qual o acionador pertence.
Nome	Esta coluna é exibida apenas se vários modelos forem selecionados no filtro. Nome do gatilho exibido como um link azul para detalhes do gatilho. Clicar no link do nome do gatilho abre o gatilho formulário de configuração . Se o gatilho for herdado de outro modelo, o nome do modelo será exibido antes do nome do gatilho, como um link cinza. Clicar no link do modelo abrirá a lista de gatilhos nesse nível de modelo.
Dados operacionais	Definição dos dados operacionais do trigger, contendo strings e macros arbitrárias que serão resolvidas dinamicamente em Monitoring → Problems.
Expressão	Exibida a expressão do gatilho. A parte do item de modelo da expressão é exibida como um link, levando ao formulário de configuração do item.
Status	O status do gatilho é exibido - Ativado ou Desativado. Ao clicar no status, você pode alterá-lo - de Ativado para Desativado (e vice-versa).
Tags	Se um acionador contiver tags, o nome e o valor da tag serão exibidos nesta coluna.

Para configurar um novo acionador, clique no botão Criar acionador na canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altera o status do gatilho para Ativado
- Desativar - altera o status do gatilho para Desativado
- Copiar - copia os gatilhos para outros hosts ou modelos
- Atualização em massa - atualiza várias propriedades para vários gatilhos de uma vez só
- Excluir - exclui os gatilhos

Para usar essas opções, marque as caixas de seleção antes do respectivo acionadores e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os gatilhos de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O ícone Filtro está disponível no canto superior direito. Clicando nele irá abrir um filtro onde você pode especificar os critérios de filtragem desejados.

Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. Somente grupos de hosts que contêm pelo menos um modelo podem ser selecionados. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
Modelos	Filtrar por um ou mais modelos. Se os grupos de hosts já estiverem selecionados acima, a seleção de modelos será limitada a esses grupos.
Nome	Filtrar pelo nome do gatilho.
Gravidade	Selecione para filtrar por uma ou várias gravidades do gatilho.
Status	Filtrar por status do gatilho.

Parâmetro	Descrição
Tags	<p>Filtrar por nome e valor da tag do acionador. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.</p> <p>Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclui os nomes de tags especificados</p> <p>Equals - inclui os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
Ou - suficiente se uma condição for atendida</p> <p>Macros e funções de macro são suportados nos campos de nome de tag e valor de tag.</p>
Herdado	Filter triggers herdados (ou não herdados) de templates vinculados.
Com dependências	Filtrar gatilhos com (ou sem) dependências.

3 Gráficos

Visão geral

A lista de gráficos personalizados para um modelo pode ser acessada em Configuration → Modelos clicando em Gráficos para o respectivo modelo.

Uma lista de gráficos existentes é exibida.

Graphs

Create graph

All templates / Template App Zabbix Server Applications 1 Items 46 Triggers 34 Graphs 6 Dashboards 1 Discovery rules Web scenarios

Filter

<input type="checkbox"/> Name ▲	Width	Height	Graph type
<input type="checkbox"/> Value cache effectiveness	900	200	Stacked
<input type="checkbox"/> Zabbix cache usage, % used	900	200	Normal
<input type="checkbox"/> Zabbix data gathering process busy %	900	200	Normal
<input type="checkbox"/> Zabbix internal process busy %	900	200	Normal
<input type="checkbox"/> Zabbix internal queues	900	200	Normal
<input type="checkbox"/> Zabbix server performance	900	200	Normal

Dados exibidos:

Coluna	Descrição
Modelo	Modelo ao qual o gráfico pertence.
Nome	<p>Esta coluna é exibida apenas se vários modelos forem selecionados no filtro.</p> <p>Nome do gráfico personalizado, exibido como um link azul para os detalhes do gráfico.</p> <p>Clicar no link do nome do gráfico abre o gráfico formulário de configuração.</p> <p>Se o gráfico for herdado de outro modelo, o nome do modelo será exibido antes do nome do gráfico, como um link cinza. Clicar no link do modelo abrirá a lista de gráficos nesse nível de modelo.</p>
Largura	A largura do gráfico é exibida.
Altura	A altura do gráfico é exibida.
Tipo de gráfico	O tipo de gráfico é exibido - Normal, Empilhado, Torta ou Explodido.

Para configurar um novo gráfico, clique no botão Criar gráfico na parte superior Esquina direita.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Copiar - copie os gráficos para outros hosts ou modelos
- Excluir - excluir os gráficos

Para usar essas opções, marque as caixas de seleção antes dos respectivos gráficos, em seguida, clique no botão desejado.

Usando filtro

Você pode filtrar gráficos por grupo de hosts e modelo. Para uma melhor pesquisa desempenho, os dados são pesquisados com macros não resolvidas.

4 Regras de descoberta

Visão geral

A lista de regras de descoberta de baixo nível para um modelo pode ser acessada de Configuração → Modelos clicando em Descoberta para o respectivo modelo.

Uma lista de regras de descoberta de baixo nível existentes é exibida. Isso é também possível ver todas as regras de descoberta independentemente do modelo, ou todas as regras de descoberta de um grupo de hosts específico alterando o filtro definições.

Discovery rules Create discovery rule

All templates / Template Server Cisco UCS SNMPv2 Items 11 Triggers 6 Graphs Dashboards Discovery rules 9 Web scenarios Filter

<input type="checkbox"/>	Template	Name	Items	Triggers	Graphs	Hosts	Key	Interval	Type	Status
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Array Controller Cache Discovery	Item prototypes 1	Trigger prototypes 2	Graph prototypes	Host prototypes	array.cache.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Array Controller Discovery	Item prototypes 2	Trigger prototypes 3	Graph prototypes	Host prototypes	array.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	FAN Discovery	Item prototypes 1	Trigger prototypes 2	Graph prototypes	Host prototypes	fan.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Physical Disk Discovery	Item prototypes 4	Trigger prototypes 2	Graph prototypes	Host prototypes	physicalDisk.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	PSU Discovery	Item prototypes 1	Trigger prototypes 2	Graph prototypes	Host prototypes	psu.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Temperature CPU Discovery	Item prototypes 1	Trigger prototypes 3	Graph prototypes	Host prototypes	temp.cpu.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Temperature Discovery	Item prototypes 4	Trigger prototypes 12	Graph prototypes	Host prototypes	temp.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Unit Discovery	Item prototypes 3	Trigger prototypes 3	Graph prototypes	Host prototypes	unit.discovery	1h	SNMP agent	Enabled
<input type="checkbox"/>	Template Server Cisco UCS SNMPv2	Virtual Disk Discovery	Item prototypes 3	Trigger prototypes 1	Graph prototypes	Host prototypes	virtualdisk.discovery	1h	SNMP agent	Enabled

0 selected Enable Disable Delete

Displaying 9 of 9 found

Dados exibidos:

Coluna	Descrição
Modelo	A regra de descoberta de modelo pertence.
Nome	Nome da regra, exibido como um link azul. Clicar no nome da regra abre a regra de descoberta de baixo nível formulário de configuração . Se a regra de descoberta é herdada de outro modelo, o nome do modelo é exibido antes do nome da regra, como um link cinza. Clicar no link do modelo abrirá a lista de regras de descoberta nesse nível de modelo.
Itens	Um link para a lista de protótipos de itens é exibido. O número de protótipos de itens existentes é exibido em cinza.
Triggers	Um link para a lista de protótipos de trigger é exibido. O número de protótipos de trigger existentes é exibido em cinza.
Gráficos	Um link para a lista de protótipos de gráficos exibidos. O número de protótipos de gráficos existentes é exibido em cinza.
Hosts	Um link para a lista de protótipos de host exibido. O número de protótipos de host existentes é exibido em cinza.
Chave	A chave do item usada para descoberta é exibida.
Intervalo	A frequência de realização da descoberta é exibida.
Tipo	O tipo de item usado para descoberta é exibido (agente Zabbix, agente SNMP, etc).
Status	O status da regra de descoberta é exibido - Ativado ou Desativado. Ao clicar no status, você pode alterá-lo - de Ativado para Desativado (e vice-versa).

Para configurar uma nova regra de descoberta de baixo nível, clique no botão Criar botão de regra de descoberta no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altere o status da regra de descoberta de baixo nível para Ativado
- Desativar - altere o status da regra de descoberta de baixo nível para Desativado
- Excluir - exclui as regras de descoberta de baixo nível

Para usar essas opções, marque as caixas de seleção antes do respectivo regras de descoberta e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas as regras de descoberta que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O ícone Filtro está disponível no canto superior direito. Clicando nele iráabrir um filtro onde poderá especificar os critérios de filtragem desejados como modelo, nome da regra de descoberta, chave de item, tipo de item etc.

All templates / VMware HypervisorItems 21TriggersGraphsDashboardsDiscovery rules 1Web scenarios

Host groups

type here to search

Select

Type

all

Status

all

Enabled

Disabled

Templates

VMware Hypervisor

type here to search

Select

Update interval

Name

Key

Keep lost resources period

Apply

Reset

Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. Somente grupos de hosts que contêm pelo menos um modelo podem ser selecionados. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
Modelos	Filtrar por um ou mais modelos.
Nome	Filtrar pelo nome da regra de descoberta.
Chave	Filtrar por chave de item de descoberta.
Tipo	Filtrar por tipo de item de descoberta.
Intervalo de atualização	Filtrar por intervalo de atualização. Não disponível para o Zabbix trapper e itens dependentes.
Manter o período de recursos perdidos	Filtrar por Manter o período de recursos perdidos.
Status	Filtrar por status da regra de descoberta (All/Enabled/Disabled).

1 Protótipos de itens

Visão geral

Nesta seção, são exibidos os itens prototypes configurados de uma regra de descoberta de baixo nível no template. Se o template estiver vinculado ao host, os itens prototypes se tornarão a base para a criação de itens reais do host **itens** durante a descoberta de baixo nível.

Item prototypes

Create item prototype

All templates / Template Module Linux filesystems...Discovery list / Mounted filesystem discovery

Item prototypes 4Trigger prototypes 4Graph prototypes 1Host prototypes

	Name	Key	Interval	History	Trends	Type	Create enabled	Discover	Tags
<input type="checkbox"/>	... {#FSNAME}: Free inodes in %	vfs.fs.inode[{#FSNAME},pfree]	1m	7d	365d	Zabbix agent	Yes	Yes	Application: Filesystem...
<input type="checkbox"/>	... {#FSNAME}: Space utilization	vfs.fs.size[{#FSNAME},pused]	1m	7d	365d	Zabbix agent	Yes	Yes	Application: Filesystem...
<input type="checkbox"/>	... {#FSNAME}: Total space	vfs.fs.size[{#FSNAME},total]	1m	7d	365d	Zabbix agent	Yes	Yes	Application: Filesystem...
<input type="checkbox"/>	... {#FSNAME}: Used space	vfs.fs.size[{#FSNAME},used]	1m	7d	365d	Zabbix agent	Yes	Yes	Application: Filesystem...

Displaying 4 of 4 found

0 selected

Create enabled

Create disabled

Mass update

Delete

Dados exibidos:

Column	Descrição
Name	O nome do item prototype será exibido como um link azul. Clicando no nome você habilitará a configuração configuration form do item prototype. Se o item prototype pertencer a um template vinculado, o nome do template será exibido como um link cinza, antes do nome do item. Clicando no link do template você abrirá a lista de item prototype vinculada ao nível do template.
Key	A chave do item prototype é exibida.
Interval	A frequência da verificação é exibida.
History	Quanto dias para manter o histórico de dado do item é exibido.
Trends	Quanto dias para manter o histórico de tendências dos itens é exibido.
Type	Tipo do item prototype é exibido (Zabbix agent, SNMP agent, verificação simples, etc).
Create enabled	Crie o item baseado neste prototype como: Sim - habilitado Não - desabilitado. Você pode alternar entre 'Sim' e 'Não', clicando neles.
Discover	Descubra o item baseado nesse prototype: Sim - descobrir Não - não descobrir. Você podem alternar entre 'Sim' e 'Não', clicando neles.
Tags	As Tags do item prototype são exibidas.

Para configurar um novo item prototype, clique no botão Criar item prototype no canto superior a direita.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Criar habilitado - crie esses itens como Habilitado
- Criar desabilitado - crie esses itens como Desabilitado
- Atualização em massa - atualização em massa desses protótipos de itens
- Excluir - exclua esses protótipos de itens

Para usar essas opções, marque as caixas de seleção antes dos respectivos protótipos de itens e clique no botão desejado.

2 Protótipos de triggers

Visão geral

Nesta seção a configuração dos protótipos triggers de uma regra de descoberta de baixo nível no template são exibidos.

Se o template estiver vinculado ao host, os protótipos triggers se tornarão a base para criação de **triggers** reais do host durante a descoberta de baixo nível.

<input type="checkbox"/>	Severity	Name ▲	Operational data	Expression	Create enabled	Discover	Tags
<input type="checkbox"/>	Average	{#FSNAME}: Disk space is critically low (used > {SVFS.FS.PUSED.MAX.CRIT:"{#FSNAME}"})%	Space used: {ITEM.LASTVALUE3} of {ITEM.LASTVALUE2} ({ITEM.LASTVALUE1})	last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},pused])>{SVFS.FS.PUSED.MAX.CRIT:"{#FSNAME}"} and ((last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},total]) -last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},used]))<5G or timeleft (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},pused],1h,100)<1d)	Yes	Yes	
<input type="checkbox"/>	Warning	{#FSNAME}: Disk space is low (used > {SVFS.FS.PUSED.MAX.WARN:"{#FSNAME}"})% Depends on: Template Module Linux filesystems by Zabbix agent: {#FSNAME}: Disk space is critically low (used > {SVFS.FS.PUSED.MAX.CRIT:"{#FSNAME}"})%	Space used: {ITEM.LASTVALUE3} of {ITEM.LASTVALUE2} ({ITEM.LASTVALUE1})	last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},pused])>{SVFS.FS.PUSED.MAX.WARN:"{#FSNAME}"} and ((last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},total]) -last (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},used]))<10G or timeleft (/Template Module Linux filesystems by Zabbix agent/vfs.fs.size[{#FSNAME},pused],1h,100)<1d)	Yes	Yes	
<input type="checkbox"/>	Average	{#FSNAME}: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.CRIT:"{#FSNAME}"})%	Free inodes: {ITEM.LASTVALUE1}	min (/Template Module Linux filesystems by Zabbix agent/vfs.fs.inode[{#FSNAME},pfree],5m)<{SVFS.FS.INODE.PFREE.MIN.CRIT:"{#FSNAME}"}	Yes	Yes	
<input type="checkbox"/>	Warning	{#FSNAME}: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.WARN:"{#FSNAME}"})% Depends on: Template Module Linux filesystems by Zabbix agent: {#FSNAME}: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.CRIT:"{#FSNAME}"})%	Free inodes: {ITEM.LASTVALUE1}	min (/Template Module Linux filesystems by Zabbix agent/vfs.fs.inode[{#FSNAME},pfree],5m)<{SVFS.FS.INODE.PFREE.MIN.WARN:"{#FSNAME}"}	Yes	Yes	

0 selected [Create enabled](#) [Create disabled](#) [Mass update](#) [Delete](#)

Displaying 4 of 4 found

Dados exibidos

Column	Descrição
Name	O nome do protótipo trigger será exibido como um link azul. Ao clicar no nome, abrirá o protótipo de trigger configuration form . Se o protótipo de trigger pertencer ao template vinculado, o nome do template será exibido antes do nome do trigger, como um link cinza. Ao clicar no link do template, abrirá a lista de protótipo de trigger no nível vinculado ao template.
Operational data	O formato dos dados operacionais do trigger é exibido, contendo strings arbitrárias e macros que resolverão dinamicamente em Monitoramento → Problema.
Create enabled	Crie o trigger baseado nesse protótipo como: Sim - habilitar Não - desabilitar. Você pode alternar entre 'Sim' e 'Não', clicando neles.
Discover	Descubra o trigger com base nesse protótipo: Sim - descobrir Não - não descobrir. Você pode alternar entre 'Sim' e 'Não', clicando neles.
Tags	Tags de protótipos de trigger são exibidos.

Para configurar um novo protótipo de trigger, clique no botão Criar protótipo de trigger no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Criar habilitado - crie essas triggers como Habilitado
- Criar desabilitado - crie essas triggers como Desabilitado
- Atualização em massa - atualização em massa desses protótipos de triggers
- Excluir - exclua esses protótipos de triggers

Para usar essas opções, marque as caixas de seleção antes dos respectivos protótipos de triggers e clique no botão necessário.

3 Protótipos de gráfico

Visão geral

Nesta seção os protótipos de gráficos configurados de uma regra de descoberta de template de baixo nível são exibidos.

Se o template estiver vinculado ao host, os protótipos de gráficos se tornarão a base para a criação de gráficos reais do host **graphs** durante a descoberta de baixo nível.

Graph prototypes

Create graph prototype

All templates / Template Module Linux filesystems...

Discovery list / Mounted filesystem discovery

Item prototypes 4

Trigger prototypes 4

Graph prototypes 1

Host prototypes

<input type="checkbox"/> Name ▲	Width	Height	Graph type	Discover
<input type="checkbox"/> {#FSNAME}: Disk space usage	600	340	Pie	Yes

0 selected

Delete

Displaying 1 of 1 found

Dado disponibilizado:

Coluna	Descrição
Name	Nome do graph prototype será exibido em um link azul. Clicando no nome, abre o graph prototype configuration form . Se o graph prototype pertencer ao link vinculado, o nome do template será exibido antes do nome do graph em um link cinza.
Width	Largura do graph prototype é exibido.
Height	Altura do graph prototype é exibido.
Type	Tipo do graph prototype é exibido - Normal, Stacked, Pie ou Exploded.
Discover	Descobrir o gráfico baseado neste protótipo: Sim - descobrir Não - não descobrir. Você pode alternar entre 'Yes' e 'No' clicando neles.

Para configurar um novo protótipo de gráfico, clique no botão Create graph prototype topo da página à direita.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Excluir - exclua esses protótipos de gráficos

Para usar essas opções, marque as caixas de seleção antes dos respectivos protótipos de gráficos e clique no botão desejado.

4 Host prototypes

Visão geral

Nesta seção, são exibidos os host prototypes configurados de uma regra de descoberta de baixo nível no template. Se os templates estiverem vinculados ao host, os host prototypes se tornarão a base para a criação de hosts reais durante uma descoberta de baixo nível **hosts**.

Host prototypes

Create host prototype

All templates / Template VM VMware

Discovery list / Discover VMware VMs

Item prototypes

Trigger prototypes

Graph prototypes

Host prototypes 1

<input type="checkbox"/> Name ▲	Templates	Create enabled	Discover	Tags
<input type="checkbox"/> {#VM.NAME}	Template VM VMware Guest	Yes	Yes	

0 selected

Create enabled

Create disabled

Delete

Displaying 1 of 1 found

Dados exibidos:

Column	Descrição
Name	O nome do host prototype será exibido em um link azul. Clicar no nome abre o formulário de configuração do host prototype. Se o host prototype pertencer a um template vinculado, o nome do template será exibido em um link cinza, antes nome do host. Clicar no link do template abrirá a lista de host prototype no nível do template.
Templates	Os templates do host prototype são exibidos.
Create enabled	Criar o host com base nesse prototype: Sim - habilitado Não - desabilitado.Você pode alternar entre 'Sim' e 'Não' clicando neles.
Discover	Descoberta do host com base nesse prototype: Sim - descobrir Não - não descobrir. Você pode alternar entre 'Sim' e 'Não' clicando neles.

Column	Descrição
Tags	As tags do host prototype serão exibidas.

Para configurar um novo host prototype, clique em Criar host prototype no canto superior direito.

Opções de edição em massa

Os botões abaixo oferecem algumas opções de edição em massa:

- Criar habilitado - criar esses hosts como Habilitado
- Criar desabilitado - criar esses hosts como Desabilitado
- Excluir - excluir esses host prototypes

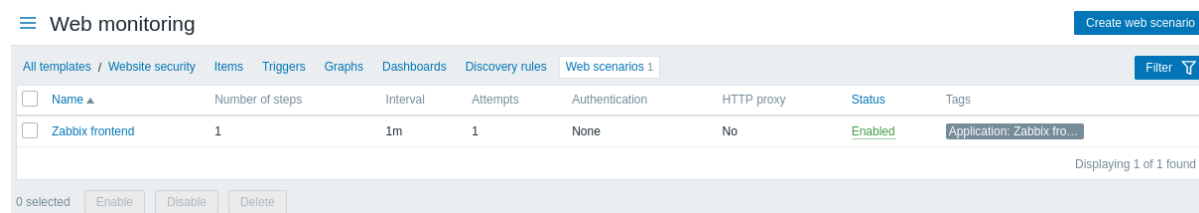
Para utilizar essas opções, assinale a caixa de seleção antes de cada host prototype, e então clique no botão necessário.

5 Cenários Web

Visão geral

A lista de cenários da Web para um modelo pode ser acessada em Configuration → Modelos clicando em Web para o respectivo modelo.

Uma lista de cenários da Web existentes é exibida.



The screenshot shows the 'Web monitoring' section of a software interface. At the top right is a 'Create web scenario' button. Below it is a navigation bar with tabs: 'All templates / Website security', 'Items', 'Triggers', 'Graphs', 'Dashboards', 'Discovery rules', and 'Web scenarios 1'. A 'Filter' button is on the right. The main table has columns: 'Name', 'Number of steps', 'Interval', 'Attempts', 'Authentication', 'HTTP proxy', 'Status', and 'Tags'. One row is visible: 'Zabbix frontend' with 1 step, 1m interval, 1 attempt, 'None' authentication, 'No' HTTP proxy, 'Enabled' status, and a tag 'Application: Zabbix fro...'. At the bottom, it says '0 selected' and has buttons for 'Enable', 'Disable', and 'Delete'. A footer note says 'Displaying 1 of 1 found'.

Dados exibidos:

Coluna	Descrição
Nome	Nome do cenário da web. Clicar no nome do cenário da web abre o [formulário de configuração] do cenário da web (/manual/web_monitoring#configuring_a_scenario). Se o cenário da web for herdado de outro modelo, o nome do modelo será exibido antes do nome do cenário da web, em cinza link. Clicar no link do modelo abrirá a lista de cenários da Web nesse nível de modelo.
Número de etapas	O número de etapas que o cenário contém.
Intervalo de atualização	Com que frequência o cenário é executado.
Tentativas	Quantas tentativas de execução das etapas do cenário da Web são realizadas.
Autenticação	O método de autenticação é exibido - Básico, NTLM em Nenhum.
HTTP proxy	Exibe o proxy HTTP ou 'Não' se não for usado.
Status	O status do cenário da Web é exibido - Ativado ou Desativado. Clicando no status você pode alterá-lo.
Tags	As tags de cenário da Web são exibidas. Até três tags (pares nome:valor) podem ser exibidas. Se houver mais tags, um link "..." é exibido que permite ver todas as tags ao passar o mouse.

Para configurar um novo cenário da web, clique em Criar cenário da web botão no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altera o status do cenário para Ativado
- Desativar - altera o status do cenário para Desativado
- Excluir - exclui os cenários da web

Para usar essas opções, marque as caixas de seleção antes do respectivo web cenários e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os cenários de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O link Filter está disponível acima da lista de cenários da web. Se você clique nele, um filtro fica disponível onde você pode filtrar cenários por grupo de hosts, modelo, status e tags.

All templates / Website securityItemsTriggersGraphsDashboardsDiscovery rulesWeb scenarios 1Filter

Host groups

type here to search

Select

Tags

.And/OrOr

tagContainsvalue

AddRemove

Templates

Website security X

type here to search

Select

Status

allEnabledDisabled

ApplyReset

1 Grupos de Host

Visão geral

Na seção Configuração → Grupos de hosts os usuários podem configurar e manter grupos de hosts. Um grupo de hosts pode conter modelos e hosts.

Uma lista de grupos de hosts existentes com seus detalhes é exibida. Vocês pode pesquisar e filtrar grupos de hosts por nome.

Host groupsCreate host groupFilter

Name

ApplyReset

<input type="checkbox"/>	Name	Hosts	Templates	Members	Info
<input type="checkbox"/>	Discovered hosts	Hosts	Templates		
<input type="checkbox"/>	Hypervisors	Hosts	Templates		
<input type="checkbox"/>	Linux servers	Hosts 4	Templates	Server1, Server2, Server3, Server4	
<input type="checkbox"/>	Templates	Hosts	Templates		
<input type="checkbox"/>	Templates/Applications	Hosts	Templates 14	Template App Apache by HTTP, Template App Apache by Zabbix agent, Template App Apache Tomcat JMX, Template App Generic Java JMX, Template App Nginx by HTTP, Template App Nginx by Zabbix agent, Template App RabbitMQ cluster by HTTP, Template App RabbitMQ cluster by Zabbix agent, Template App RabbitMQ node by HTTP, Template App RabbitMQ node by Zabbix agent, Template App Remote Zabbix proxy, Template App Remote Zabbix server, Template App Zabbix Proxy, Template App Zabbix Server	
<input type="checkbox"/>	Templates/Databases	Hosts	Templates 2	Template DB MySQL, Template DB PostgreSQL	

Dados exibidos:

Coluna	Descrição
Nome	Nome do grupo de hosts. Clicar no nome do grupo abre o grupo de hosts formulário de configuração.
Hosts	Número de hosts no grupo (exibido em cinza). Clicar em "Hosts" irá, em toda a listagem de hosts, filtrar aqueles que pertencem ao grupo.
Modelos	Número de modelos no grupo (exibidos em cinza). Clicar em "Modelos" irá, em toda a listagem de templates, filtrar aqueles que pertencem ao grupo.
Membros	Nomes dos membros do grupo. Os nomes dos modelos são exibidos em cinza, os nomes de host monitorados em azul e os nomes de host não monitorados em vermelho. Clicar em um nome abrirá o formulário de configuração de modelo/host.
Info	Informações de erro (se houver) sobre o grupo de hosts são exibidas.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar hosts - altera o status de todos os hosts do grupo para "Monitorou"
- Desabilitar hosts - altera o status de todos os hosts do grupo para "Não monitorado"
- Excluir - exclui os grupos de hosts

Para usar essas opções, marque as caixas de seleção antes do respectivo host grupos e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os grupos de hosts que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

895

2 Templates

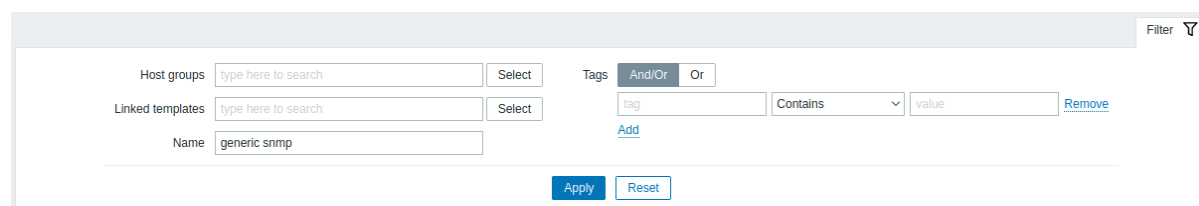
Coluna	Descrição
Nome	Nome do template. Clicar no nome do template abre o template formulário de configuração .
Hosts	Número de hosts editáveis aos quais o template está vinculado; hosts somente leitura não estão incluídos. Clicar em Hosts abrirá a lista de hosts com apenas os hosts filtrados que estão vinculados ao template.
Entidades (Itens, Acionadores, Gráficos, Painéis, Descoberta, Web)	Número das respectivas entidades no template (exibidas em cinza). Clicar no nome da entidade irá, em toda a listagem dessa entidade, filtrar aquelas que pertencem ao template.
templates vinculados	templates que estão vinculados ao template, em uma configuração aninhada onde o template herdará todas as entidades dos templates vinculados.
Vinculado aos templates	Os templates aos quais o template está vinculado (templates "filhos" que herdam todas as entidades deste template). Desde o Zabbix 5.0.3, esta coluna não inclui mais hosts.
Tags	Tags do template, com macros não resolvidas.

Para configurar um novo template, clique no botão Criar template na o canto superior direito. Para importar um template de um YAML, XML ou JSON, clique no botão Importar no canto superior direito.

Usando filtro

Você pode usar o filtro para exibir apenas os templates de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O link Filtro está disponível abaixo Criar template e Importar botões. Se você clicar nele, um filtro fica disponível onde você pode filtre templates por grupo de hosts, templates vinculados, nome e tags.



Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
Templates vinculados	Filtrar por templates vinculados diretamente.
Nome	Filtrar pelo nome do template.

Parâmetro	Descrição
Tags	<p>Filtrar por nome e valor de tag de template.</p> <p>A filtragem só é possível por tags de nível de template (não herdadas). É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas.</p> <p>Há vários operadores disponíveis para cada condição:</p> <p>Exists - inclui os nomes de tags especificados</p> <p>Equals - inclui os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas)</p> <p>Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Não existe - exclui o especificado nomes de tag</p> <p>Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas)</p> <p>Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas)</p> <p>Existem dois tipos de cálculo para condições:</p> <p>E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou
Ou - suficiente se uma condição for atendida</p>

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Exportar - exporta o modelo para um arquivo YAML, XML ou JSON
- Atualização em massa - **atualizar vários properties** para vários modelos de uma vez só
- Excluir - exclui o modelo deixando suas entidades vinculadas (itens, gatilhos etc.) com os hosts
- Excluir e limpar - exclui o modelo e suas entidades vinculadas dos anfitriões

Para usar essas opções, marque as caixas de seleção antes do respectivo modelos e, em seguida, clique no botão necessário.

3 Hosts

Visão geral

Na seção Configuração → Hosts os usuários podem configurar e manter anfitriões.

Uma lista de hosts existentes com seus detalhes é exibida.

Hosts

Create host Import

Filter

Name ▲	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status	Availability	Agent encryption	Info	Tags
Zabbix server	Items 140	Triggers 64	Graphs 27	Discovery 3	Web 1	127.0.0.1:10050		Template App Zabbix Server, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent , Template Module Linux CPU by Zabbix agent , Template Module Linux filesystems by Zabbix agent , Template Module Linux generic by Zabbix agent , Template Module Linux memory by Zabbix agent , Template Module Linux network interfaces by Zabbix agent , Template Module Zabbix agent)	Enabled	ZBX	SNMP	None	

Displaying 1 of 1 found

0 selected Enable Disable Export Mass update Delete

Dados exibidos:

Coluna	Descrição
Nome	Nome do host. Clicar no nome do host abre o host formulário de configuração .
Entidades (Itens, Acionadores, Gráficos, Descoberta, Web)	Clicar no nome da entidade exibirá itens, acionadores etc. do host. O número das respectivas entidades é exibido em cinza.

Coluna	Descrição
Interface	A interface principal do host é exibida.
Proxy	O nome do proxy será exibido, se o host for monitorado por um proxy. Esta coluna só será exibida se a opção de filtro Monitorado por estiver definida como 'Qualquer' ou 'Proxy'.
Modelos	Os modelos vinculados ao host são exibidos. Se outros modelos estiverem contidos no modelo vinculado, eles serão exibidos entre parênteses, separados por uma vírgula. Clicar no nome de um modelo abrirá seu formulário de configuração.
Status	O status do host é exibido - Enabled ou Disabled. Ao clicar no status, você pode alterá-lo. Um ícone de chave laranja antes do status do host indicar que este host está em manutenção. Os detalhes de manutenção são exibidos quando o ponteiro do mouse é posicionado sobre o ícone.
Disponibilidade	Host availability por interface configurada é exibido. Os ícones representam apenas os tipos de interface (agente Zabbix, SNMP, IPMI, JMX) que são configurado. Se você posicionar o mouse sobre o ícone, uma lista pop-up de todas as interfaces desse tipo aparecerá com os detalhes, status e erros de cada interface. A coluna está vazia para hosts sem interfaces. O status atual de todas as interfaces de um tipo é exibido pela respectiva cor do ícone: Verde - todas as interfaces disponíveis Amarelo - pelo menos uma interface disponível e pelo menos uma indisponível; outros podem ter qualquer valor, incluindo 'unknown' Red - nenhuma interface disponível Gray - pelo menos uma interface desconhecida (nenhuma indisponível) Observe que os itens ativos do agente Zabbix não afetar a disponibilidade do host.
Criptografia do agente	O status da criptografia para conexões com o host é exibido: None - sem criptografia PSK - usando chave pré-compartilhada Cert - usando certificado
Info	Informações de erro (se houver) sobre o host são exibidas.
Tags	Tags do host, com macros não resolvidas.

Para configurar um novo host, clique no botão Create host na parte superior canto direito. Para importar um host de um arquivo YAML, XML ou JSON, clique no botão Importar no canto superior direito.

Opções de atualização em massa

Ao final da lista existem botões que executam operações em massa:


- Ativar - Ativa a monitoração dos hosts selecionados
- Desativar - Desativa a monitoração dos hosts selecionados
- Exportar - Exporta os hosts selecionados para um arquivo XML
- Atualização em massa - **atualização de várias propriedades** de vários hosts de uma só vez
- Excluir - Excluir os hosts selecionados e suas entidades

Para usar estas opções, selecione os hosts e clique no botão com a ação desejada.

Usando filtro

Você pode usar o filtro para exibir apenas os hosts nos quais está interessado. Para melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de hosts. Se você clicar em ele, um filtro fica disponível onde você pode filtrar hosts por grupo de hosts, modelos vinculados, nome, DNS, IP, número da porta, se forem monitorados por servidor ou por proxy, nome de proxy e tags.

Filter 

Host groups

Templates

Name

DNS

IP

Port

Monitored by Any Server Proxy

Proxy

Tags And/Or Or

Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
Modelos	Filtrar por modelos vinculados.
Nome	Filtrar por nome de host visível.
DNS	Filtrar por nome DNS.
IP	Filtrar por endereço IP.
Porta	Filtrar por número de porta.
Monitorado por	Filtrar hosts que são monitorados apenas pelo servidor, somente proxy ou ambos.
Proxy	Filtrar hosts que são monitorados pelo proxy especificado aqui.
Tags	Filtrar por nome e valor de tag de host. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas. Há vários operadores disponíveis para cada condição: Exists - inclui os nomes de tags especificados Equals - inclui os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas) Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas) Não existe - exclui o especificado nomes de tag Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas) Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas) Existem dois tipos de cálculo para condições: E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou Ou - suficiente se uma condição for atendida

Lendo a disponibilidade do host

Os ícones de disponibilidade do host refletem o status atual da interface do host em Servidor Zabbix. Portanto, no frontend:

- Se você desabilitar um host, os ícones de disponibilidade não serão ativados imediatamente cinza (status desconhecido), pois o servidor precisa sincronizar o a configuração muda primeiro;
- Se você habilitar um host, os ícones de disponibilidade não serão ativados imediatamente verde (disponível), pois o servidor precisa sincronizar o alterações de configuração e comece a sondar o host primeiro.

Status de interface desconhecido

O servidor Zabbix determina um status desconhecido para o agente correspondente interface (Zabbix, SNMP, IPMI, JMX) se:

- não há itens habilitados na interface (eles foram removidos ou Desativado);
- existem apenas itens de agente Zabbix ativos;
- não há pollers para esse tipo de interface (por exemplo, StartPollers=0);
- o host está desabilitado;
- o host está configurado para ser monitorado por proxy, um proxy diferente ou por servidor se foi monitorado por procuração;
- o host é monitorado por um proxy que parece estar offline (sem atualizações recebido do proxy durante o intervalo máximo de pulsação - 1 hora).

A configuração da disponibilidade da interface como desconhecida é feita após o servidor sincronização de cache de configuração. Restaurando a disponibilidade da interface (disponível/indisponível) em hosts monitorados por proxies é feito após sincronização de cache de configuração de proxy.

Veja também mais detalhes sobre a interface do host **inacessibilidade**.

2 Itens

Visão geral

A lista de itens para um host pode ser acessada em Configuration → Hosts por clicando em Itens para o respectivo host.

Uma lista de itens existentes é exibida.

Items

Create item

All hosts / Zabbix server Enabled ZBX SNMP IPMI Items 146 Triggers 67 Graphs 27 Discovery rules 3 Web scenarios 1

Filter

<input type="checkbox"/>	Name	Triggers	Key	Interval	History	Trends	Type	Status	Tags	Info
<input type="checkbox"/>	... Template Module Zabbix agent: Host name of Zabbix agent running		agent.hostname	1h	7d		Zabbix agent (active)	Enabled	App: 1 App: 2 App: 3 ...	
<input type="checkbox"/>	... Template Module Zabbix agent: Zabbix agent ping		agent.ping	1m	1d	365d	Zabbix agent	Enabled	Application: Monitorin...	
<input type="checkbox"/>	... Template Module Zabbix agent: Version of Zabbix agent running		agent.version	1h	7d		Zabbix agent	Enabled	Application: Monitorin...	
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent: Maximum number of open file descriptors	Triggers 1	kernel.maxfiles	1h	7d	365d	Zabbix agent	Enabled	Application: General	
<input type="checkbox"/>	... Template Module Linux generic by Zabbix agent: Maximum number of processes	Triggers 2	kernel.maxproc	1h	7d	365d	Zabbix agent	Enabled	Application: General	
<input type="checkbox"/>	... A Interface \$1: Inbound packets, compressed		net.if.in["enp4s0",compressed]	3m	7d	365d	Zabbix agent	Enabled	Application: Interface ...	
<input type="checkbox"/>	... Network interface discovery: Interface enp4s0: Inbound packets discarded		net.if.in["enp4s0",discarded]	3m	7d	365d	Zabbix agent	Enabled	Application: Interface ...	

Dados exibidos:

Coluna	Descrição
Menu do item	Clique no ícone de três pontos para abrir o menu do item específico com estas opções: Últimos dados - veja os dados mais recentes do item Criar acionador - crie um acionador com base neste item Acionadores - clique para ver uma lista com links para o acionador já configurado deste item Criar item dependente - crie um item dependente para este item Criar regra de descoberta dependente - criar uma regra de descoberta dependente para este item
Host	Host do item.
Nome	Esta coluna é exibida apenas se vários hosts forem selecionados no filtro. Nome do item exibido como um link azul para os detalhes do item. Clicar no link do nome do item abre o item formulário de configuração . Se o item do host pertencer a um modelo, o nome do modelo será exibido antes do nome do item como um link cinza. Clicar no link do modelo abrirá a lista de itens no nível do modelo. Se o item tiver sido criado a partir de um protótipo de item, seu nome será precedido pelo nome da regra de descoberta de baixo nível, em laranja. Clicar no nome da regra de descoberta abrirá a lista de protótipos de itens.
Triggers	Mover o mouse sobre Triggers exibirá uma caixa de informações exibindo os gatilhos associados ao item.
Chave	O número de gatilhos é exibido em cinza.
Intervalo	A chave do item é exibida. A frequência da verificação é exibida. Observe que os itens passivos também podem ser verificados imediatamente pressionando o botão Verificar agora .
Histórico	Por quantos dias o histórico de dados do item será mantido é exibido.
Trends	É exibido por quantos dias o histórico de tendências do item será mantido.
Tipo	O tipo de item é exibido (agente Zabbix, agente SNMP, verificação simples, etc).

Coluna	Descrição
Status	O status do item é exibido - Ativado, Desativado ou Não suportado. Você pode alterar o status clicando nele - de Ativado para Desativado (e vice-versa); de Não suportado para Desativado (e vice-versa).
Tags	As tags de item são exibidas. Até três tags (pares nome:valor) podem ser exibidas. Se houver mais tags, um link "..." é exibido que permite ver todas as tags ao passar o mouse.
Info	Se o item estiver funcionando corretamente, nenhum ícone será exibido nesta coluna. Em caso de erros, um ícone quadrado com a letra "i" é exibido. Passe o mouse sobre o ícone para ver uma dica de ferramenta com a descrição do erro.

Para configurar um novo item, clique no botão Criar item na parte superior Esquina direita.

Atualização em massa

Ao final da lista existem botões para atualização em massa dos registros:

- Ativar - modifica o status dos itens selecionados para Ativo
- Desativar - modifica o status dos itens selecionados para Inativo
- Copiar - copia o item para outro host ou template
- Atualização em massa - atualização de várias propriedades de vários itens de uma só vez
- Limpar histórico - limpa o histórico e as médias dos itens selecionados
- Excluir - exclui os itens selecionados

Para utilizar estes botões, selecione os itens e clique no botão com a ação desejada.

Filtro

Uma lista de itens pode conter muitos itens e o uso do filtro pode ser realmente necessário.

O link para o Filtro está situado logo abaixo da barra de título. Ao clicar nele é apresentado o formulário com as opções de filtro.

The screenshot shows the Zabbix Filter interface. At the top, there's a navigation bar with links like 'All hosts', 'Remote proxy: New host', 'Enabled', 'ZBX', 'SNMP', 'JMX', 'IPMI', 'Applications 11', 'Items 41', 'Triggers 18', 'Graphs 7', 'Discovery rules 2', and 'Web scenarios 1'. A 'Filter' button is on the right. Below this is a form with various filter options: 'Host group' (search and select), 'Host' (search and select), 'Application' (select), 'Name' (text input), 'Key' (text input), 'Type' (dropdown), 'Type of information' (dropdown), 'State' (dropdown), 'Update interval' (text input), 'History' (text input), 'Trends' (text input), 'Status' (dropdown), 'Triggers' (dropdown), 'Template' (dropdown), and 'Discovery' (dropdown). There are 'Apply' and 'Reset' buttons. Below the form, a message says 'Subfilter affects only filtered data'. The main section shows a list of filtered items, categorized by 'APPLICATIONS', 'TYPES', 'TYPE OF INFORMATION', 'STATUS', 'TEMPLATE', 'WITH TRIGGERS', 'DISCOVERY', and 'HISTORY'. Each category has a list of items with counts, such as 'CPU 13', 'Filesystems 5', 'General 5', 'Memory 5', 'Network interfaces 2', 'OS 8', 'Performance 13', 'Processes 1', 'Security 2', 'Zabbix agent 3', 'Zabbix agent 38', 'Zabbix agent (active) 1', 'Zabbix trapper 1', 'Character 5', 'Numeric (float) 14', 'Numeric (unsigned) 21', 'Disabled +1', 'Enabled +2', 'Not Templated items 9', 'Templated items 31', 'Without triggers 24', 'With triggers 16', 'Discovered 7', 'Regular 33', '7d 38', '3m 2', '30s 1', '1m 26', '10m 2', '1h 10'.

O bloco **Subfiltro** abaixo do filtro oferece restrições adicionais para os dados que já foram filtrados. Você pode selecionar grupos de itens com características em comum. Se você clicar em um grupo este será realçado e somente os itens que contiverem aquela característica serão exibidos na listagem.

2 Triggers

A lista de triggers para um host pode ser acessada em Configuration → Hosts clicando em Triggers para o respectivo host.

All hosts / Zabbix server		Enabled	ZBX	SNMP	IPMI	JMX	Items 142	Triggers 67	Graphs 27	Discovery rules 3	Web scenarios 1	Filter
Severity	Value	Name	Operational data	Expression	Status	Info	Tags					
<input type="checkbox"/>	Average	OK	Mounted filesystem discovery: /: Disk space is critically low (used > {SVFS.FS.PUSED.MAX.CRIT:"7"}%)	Space used: {ITEM.LASTVALUE3} of {ITEM.LASTVALUE2} ({ITEM.LASTVALUE1})	<code>last(/Zabbix server/vfs.fs.size[/,pused])>{SVFS.FS.PUSED.MAX.CRIT:"7"} and ((last(/Zabbix server/vfs.fs.size[/,total])-last(/Zabbix server/vfs.fs.size[/,used]))<5G or timeleft(/Zabbix server/vfs.fs.size[/,pused],1h,100)<1d)</code>	Enabled						
<input type="checkbox"/>	Warning	OK	Mounted filesystem discovery: /: Disk space is low (used > {SVFS.FS.PUSED.MAX.WARN:"7"}%) Depends on: Zabbix server: /: Disk space is critically low (used > {SVFS.FS.PUSED.MAX.CRIT:"7"}%)	Space used: {ITEM.LASTVALUE3} of {ITEM.LASTVALUE2} ({ITEM.LASTVALUE1})	<code>last(/Zabbix server/vfs.fs.size[/,pused])>{SVFS.FS.PUSED.MAX.WARN:"7"} and ((last(/Zabbix server/vfs.fs.size[/,total])-last(/Zabbix server/vfs.fs.size[/,used]))<10G or timeleft(/Zabbix server/vfs.fs.size[/,pused],1h,100)<1d)</code>	Enabled						
<input type="checkbox"/>	Average	OK	Mounted filesystem discovery: /: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.CRIT:"7"}%)	Free inodes: {ITEM.LASTVALUE1}	<code>min(/Zabbix server/vfs.fs.inode[/,pfree],5m)<{SVFS.FS.INODE.E.PFREE.MIN.CRIT:"7"}</code>	Enabled						
<input type="checkbox"/>	Warning	OK	Mounted filesystem discovery: /: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.WARN:"7"}%) Depends on: Zabbix server: /: Running out of free inodes (free < {SVFS.FS.INODE.PFREE.MIN.CRIT:"7"}%)	Free inodes: {ITEM.LASTVALUE1}	<code>min(/Zabbix server/vfs.fs.inode[/,pfree],5m)<{SVFS.FS.INODE.E.PFREE.MIN.WARN:"7"}</code>	Enabled						
<input type="checkbox"/>	Information	OK	Template Module Linux generic by Zabbix agent: /etc/passwd has been changed Depends on: Zabbix server: Operating system description has changed Zabbix server: System name has changed (new name: {ITEM.VALUE})		<code>(last(/Zabbix server/vfs.file.cksum[/etc/passwd],#1)<last(/Zabbix server/vfs.file.cksum[/etc/passwd],#2))>0</code>	Enabled						

Dados exibidos:

Coluna	Descrição
Gravidade	A gravidade do acionador é exibida pelo nome e pela cor de fundo da célula.
Valor	O valor do triggers é exibido: OK - o triggers está no estado OK PROBLEM - o triggers está no estado Problema
Host	Host do acionador.
Nome	Esta coluna é exibida apenas se vários hosts forem selecionados no filtro. Nome do triggers, exibido como um link azul para detalhes do triggers. Clicar no link do nome do triggers abre o triggers formulário de configuração . Se o acionador do host pertencer a um modelo, o nome do modelo será exibido antes do nome do acionador, como um link cinza. Clicar no link do modelo abrirá a lista de acionadores no nível do modelo. Se o acionador tiver sido criado a partir de um protótipo de acionador, seu nome será precedido pelo nome da regra de descoberta de baixo nível, em laranja. Clicar no nome da regra de descoberta abrirá a lista de protótipos do acionador.
Dados operacionais	Definição dos dados operacionais do trigger, contendo strings e macros arbitrárias que serão resolvidas dinamicamente em Monitoring → Problems.
Expressão	Exibida a expressão do triggers. A parte do item do host da expressão é exibida como um link, levando ao formulário de configuração do item.
Status	O status do triggers é exibido - Ativado, Desativado ou Desconhecido. Ao clicar no status você pode alterá-lo - de Ativado para Desativado (e vice-versa); de Desconhecido para Desabilitado (e vice-versa). Problemas de um triggers desabilitado não são mais exibidos no frontend, mas não são excluídos.
Info	Se tudo estiver funcionando corretamente, nenhum ícone será exibido nesta coluna. Em caso de erros, um ícone quadrado com a letra "i" é exibido. Passe o mouse sobre o ícone para ver uma dica de ferramenta com a descrição do erro.
Tags	Se um acionador contiver tags, o nome e o valor da tag serão exibidos nesta coluna.

Para configurar um novo acionador, clique no botão Criar acionador na canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altera o status do gatilho para Ativado.
- Desativar - altera o status do gatilho para Desativado.
- Copiar - copia os gatilhos para outros hosts ou modelos.
- Atualização em massa - atualiza várias propriedades para vários gatilhos de uma vez só.
- Excluir - exclui os gatilhos.

Para usar essas opções, marque as caixas de seleção antes do respectivo acionadores e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os gatilhos de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O ícone Filtro está disponível no canto superior direito. Clicando nele irá abrir um filtro onde você pode especificar os critérios de filtragem desejados.

Filter

Host groups

type here to search

Select

Hosts

type here to search

Select

Name

Severity

☐ Not classified

☐ Warning

☐ High

☐ Information

☐ Average

☐ Disaster

State

all

Normal

Unknown

Status

all

Enabled

Disabled

Value

all

Ok

Problem

Tags

And/Or

Or

tag

Contains

value

Remove

Add

Inherited

all

Yes

No

Discovered

all

Yes

No

With dependencies

all

Yes

No

Apply

Reset

Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados. Grupos de hosts contendo apenas modelos não podem ser selecionados.
Hosts	Filtrar por um ou mais hosts. Se os grupos de hosts já estiverem selecionados acima, a seleção de hosts é limitada a esses grupos.
Nome	Filtrar pelo nome do gatilho.
Gravidade	Selecione para filtrar por uma ou várias gravidades do gatilho.
Estado	Filtrar por estado de disparo.
Status	Filtrar por status do gatilho.
Valor	Filtrar por valor de disparo.
Tags	Filtrar por nome e valor da tag do acionador. É possível incluir e excluir tags e valores de tag específicos. Várias condições podem ser definidas. A correspondência de nomes de tags sempre diferencia maiúsculas de minúsculas. Há vários operadores disponíveis para cada condição: Exists - inclua os nomes de tags especificados Equals - inclua os nomes de tags especificados e valores (diferencia maiúsculas de minúsculas) Contém - inclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas) Não existe - exclui o especificado nomes de tag Não é igual - exclui os nomes e valores de tag especificados (diferencia maiúsculas de minúsculas) Não contém - exclui os nomes de tag especificados onde os valores de tag contêm a string inserida (correspondência de substring, não diferencia maiúsculas de minúsculas) Existem dois tipos de cálculo para condições: E/ou - todas as condições devem ser atendidas, condições com o mesmo nome de tag serão agrupadas pela condição Ou Ou - suficiente se uma condição for atendida Macros e funções de macro são suportados tanto nos campos de nome de tag quanto de valor de tag.
Herdado	Filter triggers herdados (ou não herdados) de um template.
Descobertos	Acionadores de filtro descobertos (ou não descobertos) pela descoberta de baixo nível.
Com dependências	Filtrar gatilhos com (ou sem) dependências.

Visão geral

A lista de gráficos personalizados para um host pode ser acessada em Configuration → Hosts clicando em Graphs para o respectivo host.

Uma lista de gráficos existentes é exibida.

Graphs

Create graph

All hosts / Zabbix server Enabled ZBX SNMP IPMI JMX Items 151 Triggers 68 Graphs 30 Discovery rules 3 Web scenarios 1

Filter

<input type="checkbox"/>	Name ▲	Width	Height	Graph type	Info
<input type="checkbox"/>	Mounted filesystem discovery: /: Disk space usage	600	340	Pie	
<input type="checkbox"/>	Template Module Linux CPU by Zabbix agent: CPU jumps	900	200	Normal	
<input type="checkbox"/>	Template Module Linux CPU by Zabbix agent: CPU usage	900	200	Stacked	
<input type="checkbox"/>	Template Module Linux CPU by Zabbix agent: CPU utilization	900	200	Normal	
<input type="checkbox"/>	Network interface discovery: Interface enp4s0: Network traffic	900	200	Normal	
<input type="checkbox"/>	Network interface discovery: Interface ppp0: Network traffic	900	200	Normal	<div>i</div>
<input type="checkbox"/>	Network interface discovery: Interface wlp3s0: Network traffic	900	200	Normal	
<input type="checkbox"/>	Template Module Linux memory by Zabbix agent: Memory usage	900	200	Normal	
<input type="checkbox"/>	Template Module Linux memory by Zabbix agent: Memory utilization	900	200	Normal	
<input type="checkbox"/>	Template Module Linux generic by Zabbix agent: Processes	900	200	Normal	
<input type="checkbox"/>	Block devices discovery: sda: Disk average waiting time	900	200	Normal	
<input type="checkbox"/>	Block devices discovery: sda: Disk read/write rates	900	200	Normal	

Dados exibidos:

Coluna	Descrição
Nome	Nome do gráfico personalizado, exibido como um link azul para os detalhes do gráfico. Clicar no link do nome do gráfico abre o gráfico formulário de configuração . Se o gráfico do host pertencer a um modelo, o nome do modelo será exibido antes do nome do gráfico, como um link cinza. Clicar no link do modelo abrirá a lista de gráficos no nível do modelo. Se o gráfico tiver sido criado a partir de um protótipo de gráfico, seu nome será precedido pelo nome da regra de descoberta de baixo nível, em laranja. Clicar no nome da regra de descoberta abrirá a lista de protótipos do gráfico.
Largura	A largura do gráfico é exibida.
Altura	A altura do gráfico é exibida.
Tipo de gráfico	O tipo de gráfico é exibido - Normal, Empilhado, Torta ou Explodido.
Info	Se o gráfico estiver funcionando corretamente, nenhum ícone será exibido nesta coluna. Em caso de erros, um ícone quadrado com a letra "i" é exibido. Passe o mouse sobre o ícone para ver uma dica de ferramenta com a descrição do erro.

Para configurar um novo gráfico, clique no botão Criar gráfico na parte superior Esquina direita.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Copiar - copie os gráficos para outros hosts ou modelos
- Excluir - excluir os gráficos

Para usar essas opções, marque as caixas de seleção antes dos respectivos gráficos, em seguida, clique no botão necessário.

Usando filtro

Você pode filtrar gráficos por grupo de hosts e host. Para uma melhor pesquisa desempenho, os dados são pesquisados com macros não resolvidas.

4 Regras de descoberta

Visão geral

A lista de regras de descoberta de baixo nível para um host pode ser acessada em Configuração → Hosts clicando em Descoberta para o respectivo hospedeiro.

Uma lista de regras de descoberta de baixo nível existentes é exibida. Isso é também possível ver todas as regras de descoberta independentemente do host, ou todas regras de descoberta de um grupo de hosts específico alterando o filtro definições.

Discovery rules

Create discovery rule

All hosts / Zabbix server Enabled ZBX SNMP IPMI JMX Items 151 Triggers 68 Graphs 30 Discovery rules 3 Web scenarios 1

Filter

<input type="checkbox"/>	Host	Name	Items	Triggers	Graphs	Hosts	Key	Interval	Type	Status	Info
<input type="checkbox"/>	Zabbix server	Template Module Linux block devices by Zabbix agent: Get /proc/diskstats: Block devices discovery	Item prototypes 8	Trigger prototypes 1	Graph prototypes 3	Host prototypes	vfs.dev.discovery		Dependent item	Enabled	
<input type="checkbox"/>	Zabbix server	Template Module Linux filesystems by Zabbix agent: Mounted filesystem discovery	Item prototypes 4	Trigger prototypes 4	Graph prototypes 1	Host prototypes	vfs.fs.discovery	1h	Zabbix agent	Enabled	
<input type="checkbox"/>	Zabbix server	Template Module Linux network interfaces by Zabbix agent: Network interface discovery	Item prototypes 8	Trigger prototypes 3	Graph prototypes 1	Host prototypes	net.if.discovery	1h	Zabbix agent	Enabled	

0 selected Enable Disable Execute now Delete

Displaying 3 of 3 found

Dados exibidos:

Coluna	Descrição
Host	O nome do host visível é exibido.
Nome	Na ausência de um nome de host visível, o nome do host técnico é exibido. Nome da regra, exibido como um link azul. Clicar no nome da regra abre a regra de descoberta de baixo nível formulário de configuração . Se a regra de descoberta pertence a um modelo, o nome do modelo é exibido antes do nome da regra, como um link cinza. Clicar no link do modelo abrirá a lista de regras no nível do modelo.
Itens	Um link para a lista de protótipos de itens é exibido.
Triggers	O número de protótipos de itens existentes é exibido em cinza. Um link para a lista de protótipos de trigger é exibido.
Gráficos	O número de protótipos de trigger existentes é exibido em cinza. Um link para a lista de protótipos de gráficos é exibido.
Hosts	O número de protótipos de gráficos existentes é exibido em cinza. Um link para a lista de protótipos de host é exibido.
Chave	O número de protótipos de host existentes é exibido em cinza.
Intervalo	A chave do item usada para descoberta é exibida. A frequência de execução da descoberta é exibida.
Tipo	Observe que a descoberta também pode ser realizada imediatamente pressionando o botão Verificar agora abaixo da lista. O tipo de item usado para descoberta é exibido (agente Zabbix, agente SNMP, etc).
Status	O status da regra de descoberta é exibido - Ativado, Desativado ou Não suportado. Ao clicar no status você pode alterá-lo - de Ativado para Desativado (e vice-versa); de Não suportado para Desativado (e vice-versa).
Info	Se tudo estiver bem, nenhum ícone será exibido nesta coluna. Em caso de erros, um ícone quadrado com a letra "i" é exibido. Passe o mouse sobre o ícone para ver uma dica de ferramenta com a descrição do erro.

Para configurar uma nova regra de descoberta de baixo nível, clique no botão Criar botão de regra de descoberta no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altera o status da regra de descoberta de baixo nível para Ativado.
- Desativar - altera o status da regra de descoberta de baixo nível para Desabilitado.
- Verifique agora - execute a descoberta com base nas regras de descoberta imediatamente. Veja **mais detalhes**. Observe que ao realizar a descoberta imediatamente, a configuração cache não é atualizado, portanto, o resultado não refletirá muito recente alterações na configuração da regra de descoberta.
- Excluir - exclui as regras de descoberta de baixo nível.

Para usar essas opções, marque as caixas de seleção antes do respectivo regras de descoberta e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas as regras de descoberta que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de regras de descoberta. Se você clique nele, um filtro fica disponível onde você pode filtrar a descoberta regras por grupo de hosts, host, nome, chave de item, tipo de item e outros parâmetros.

All hosts / Zabbix server

EnabledZBXSNMPIPMIJMX

Items 151

Triggers 68

Graphs 30

Discovery rules 3

Web scenarios 1

Filter

Host groups

type here to search

Select

Hosts

Zabbix server

type here to search

Select

Name

Key

Type

all

Update interval

Keep lost resources period

State

allNormalNot supported

Status

allEnabledDisabled

ApplyReset

Parâmetro	Descrição
Grupos de hosts	Filtrar por um ou mais grupos de hosts. A especificação de um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.
Hosts	Filtrar por um ou mais hosts.
Nome	Filtrar pelo nome da regra de descoberta.
Chave	Filtrar por chave de item de descoberta.
Tipo	Filtrar por tipo de item de descoberta.
Intervalo de atualização	Filtrar por intervalo de atualização.
Manter o período de recursos perdidos	Não disponível para o Zabbix trapper e itens dependentes. Filtrar por Manter o período de recursos perdidos.
SNMP OID	Filtrar por SNMP OID.
Estado	Disponível apenas se SNMP agent for selecionado como tipo. Filtrar por estado da regra de descoberta (Todos/Normal/Não suportado).
Status	Filtrar por status da regra de descoberta (All/Enabled/Disabled).

1 Protótipos de itens

Overview

In this section the item prototypes of a low-level discovery rule on the host are displayed. Item prototypes are the basis of real host **items** that are created during low-level discovery.

Item prototypes

Create item prototype

All hosts / Zabbix server

EnabledZBXSNMPIPMI

Discovery list / Network interface discovery

Item prototypes 8

Trigger prototypes 3

Graph prototypes 1

Host prototypes

	Name ▲	Key	Interval	History	Trends	Type	Create enabled	Discover	Tags
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Bits received	net.if.in["{#IFNAME}"]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Bits sent	net.if.out["{#IFNAME}"]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Inbound packets discarded	net.if.in["{#IFNAME}",dropped]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Inbound packets with errors	net.if.in["{#IFNAME}",errors]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Interface type	vfs.file.contents["/sys/class/net/{#IFNAME}/type"]	1h	7d	0d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Operational status	vfs.file.contents["/sys/class/net/{#IFNAME}/operstate"]	1m	7d	0	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Outbound packets discarded	net.if.out["{#IFNAME}",dropped]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...
<input type="checkbox"/>	... Template Module Linux network interfaces by Zabbix agent: Interface {#IFNAME}: Outbound packets with errors	net.if.out["{#IFNAME}",errors]	3m	7d	365d	Zabbix agent	Yes	Yes	Application: Interface {...

Displaying 8 of 8 found

0 selectedCreate enabledCreate disabledMass updateDelete

Displayed data:

Displayed data:

Column	Description
Name	<p>Name of the trigger prototype, displayed as a blue link.</p> <p>Clicking on the name opens the trigger prototype configuration form.</p> <p>If the trigger prototype belongs to a linked template, the template name is displayed before the trigger name, as a gray link. Clicking on the template link will open the trigger prototype list on the linked template level.</p>
Operational data	Format of the operational data of the trigger is displayed, containing arbitrary strings and macros that will resolve dynamically in Monitoring → Problems.
Create enabled	<p>Create the trigger based on this prototype as:</p> <p>Yes - enabled</p> <p>No - disabled. You can switch between 'Yes' and 'No' by clicking on them.</p>
Discover	<p>Discover the trigger based on this prototype:</p> <p>Yes - discover</p> <p>No - do not discover. You can switch between 'Yes' and 'No' by clicking on them.</p>
Tags	Tags of the trigger prototype are displayed.

To configure a new trigger prototype, click on the Create trigger prototype button at the top right corner.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Criar habilitado - crie essas triggers como Habilitado
- Criar desabilitado - crie essas triggers como Desabilitado
- Atualização em massa - atualização em massa desses protótipos de triggers
- Excluir - exclua esses protótipos de triggers

Para usar essas opções, marque as caixas de seleção antes dos respectivos protótipos de triggers e clique no botão necessário.

3 Protótipos de gráfico

Overview

In this section the graph prototypes of a low-level discovery rule on the host are displayed. Graph prototypes are the basis of real host **graphs** that are created during low-level discovery.

Graph prototypes

Create graph prototype

All hosts / Zabbix server

Enabled

ZBX SNMP IPMI

Discovery list / Network interface discovery

Item prototypes 8

Trigger prototypes 3

Graph prototypes 1

Host prototypes

<input type="checkbox"/> Name ▲	Width	Height	Graph type	Discover
<input type="checkbox"/> Template Module Linux network interfaces by Zabbix agent: Interface (#IFNAME): Network traffic	900	200	Normal	Yes

Displaying 1 of 1 found

0 selected

Delete

Displayed data:

Column	Description
Name	<p>Name of the graph prototype, displayed as a blue link.</p> <p>Clicking on the name opens the graph prototype configuration form.</p> <p>If the graph prototype belongs to a linked template, the template name is displayed before the graph name, as a gray link. Clicking on the template link will open the graph prototype list on the linked template level.</p>
Width	Width of the graph prototype is displayed.
Height	Height of the graph prototype is displayed.
Type	Type of the graph prototype is displayed - Normal, Stacked, Pie or Exploded.

Column	Description
Discover	Discover the graph based on this prototype: Yes - discover No - do not discover. You can switch between 'Yes' and 'No' by clicking on them.

To configure a new graph prototype, click on the Create graph prototype button at the top right corner.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Excluir - exclua esses protótipos de gráficos

Para usar essas opções, marque as caixas de seleção antes dos respectivos protótipos de gráficos e clique no botão desejado.

4 Host prototypes

Visão geral

Nesta seção, os host prototypes de uma regra de descoberta de baixo nível no host são exibidos. Os host prototypes são a base para hosts reais que são criados durante uma descoberta de baixo nível **hosts**.

≡

Host prototypes

Create host prototype

All hosts / Server1 Enabled ZBX

Discovery list / Discover VMware VMs

Item prototypes

Trigger prototypes

Graph prototypes

Host prototypes 1

<input type="checkbox"/>	Name ▲	Templates	Create enabled	Discover	Tags
<input type="checkbox"/>	Template VM VMware: {#VM.NAME}	Template VM VMware Guest	Yes	Yes	

0 selected

Create enabled

Create disabled

Delete

Displaying 1 of 1 found

Dados exibidos:

Column	Descrição
Name	Nome do host prototype, exibido em um link azul. Clicar no nome abre o formulário de configuração do host prototype. Se o host prototype pertencer a um template vinculado, o nome do template será exibido em um link cinza, antes do nome do host. Clicar no link do template abrirá a lista do host prototype no nível do vinculado.
Templates	Os templates do host prototype serão exibidos.
Create enabled	Criar o host com base neste prototype como: Sim - habilitado Não - desabilitado. Você pode alternar entre 'Sim' e 'Não' clicando neles.
Discover	Descobrir o host com base neste prototype: Sim - descobrir Não - não descobrir. Você pode alternar entre 'Sim' e 'Não' clicando neles.
Tags	As Tags do host prototype serão exibidas.

Para configurar um novo host prototype, clicar no botão Create host prototype do canto superior direito.

Opções de edição em massa

Os botões abaixo oferecem algumas opções de edição em massa:

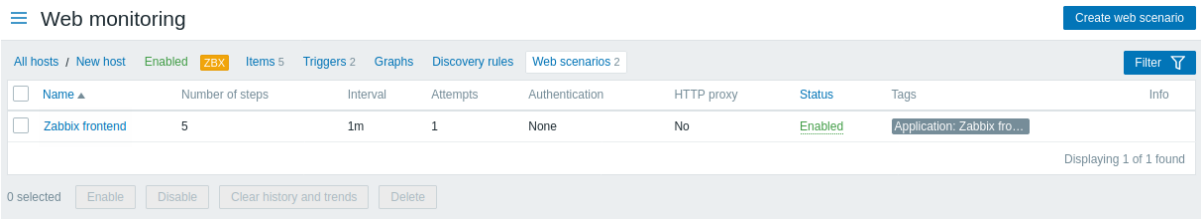
- Criar habilitado - criar estes hosts como Habilitado
- Criar desabilitado - criar estes hosts como Desabilitado
- Excluir - excluir estes host prototypes

Para utilizar essas opções, assinala a caixa de seleção antes dos host prototypes respectivos, e em seguida, clique no botão necessário.

5 Cenários Web

Visão geral

A lista de cenários da web para um host pode ser acessada em Configuration → Hosts clicando em Web para o respectivo host. Uma lista de cenários da Web existentes é exibida.



Dados exibidos:

Coluna	Descrição
Nome	Nome do cenário da web. Clicar no nome do cenário da Web abre o [formulário de configuração] do cenário da Web (/manual/web_monitoring#configuring_a_scenario). Se o cenário da Web do host pertencer a um modelo, o nome do modelo será exibido antes do nome do cenário da Web como um link cinza . Clicar no link do modelo abrirá a lista de cenários da Web no nível do modelo.
Número de etapas	O número de etapas que o cenário contém.
Intervalo de atualização	Com que frequência o cenário é executado.
Tentativas	Quantas tentativas de execução das etapas do cenário da Web são realizadas.
Autenticação	O método de autenticação é exibido - Básico, NTLM ou Nenhum.
HTTP proxy	Exibe o proxy HTTP ou 'Não' se não for usado.
Status	O status do cenário da Web é exibido - Ativado ou Desativado. Clicando no status você pode alterá-lo.
Tags	As tags de cenário da Web são exibidas. Até três tags (pares nome:valor) podem ser exibidas. Se houver mais tags, um link "..." é exibido que permite ver todas as tags ao passar o mouse.
Info	Se tudo estiver funcionando corretamente, nenhum ícone será exibido nesta coluna. Em caso de erros, um ícone quadrado com a letra "i" é exibido. Passe o mouse sobre o ícone para ver uma dica de ferramenta com a descrição do erro.

Para configurar um novo cenário da web, clique em Criar cenário da web botão no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

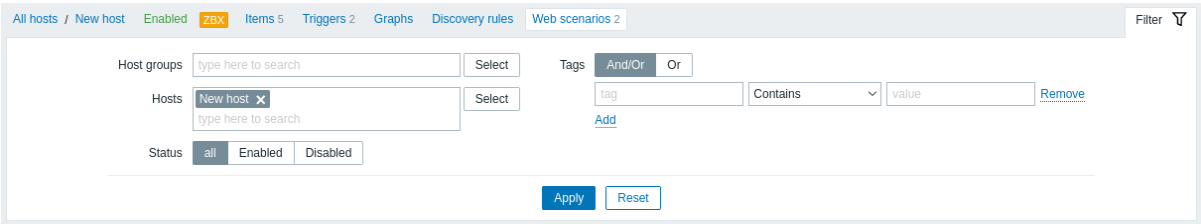
- Ativar - altera o status do cenário para Ativado
- Desativar - altera o status do cenário para Desativado
- Limpar histórico - limpar histórico e dados de tendência para os cenários
- Excluir - exclui os cenários da web

Para usar essas opções, marque as caixas de seleção antes do respectivo web cenários e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os cenários de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O link Filter está disponível acima da lista de cenários da web. Se você clique nele, um filtro fica disponível onde você pode filtrar cenários por grupo de hosts, host, status e tags.



4 Manutenção

Visão geral

Na seção Configuração → Manutenção os usuários podem configurar e manter períodos de manutenção para hosts.

Uma lista de períodos de manutenção existentes com seus detalhes é exibido.

☰ Maintenance periods

Create maintenance period

<input type="checkbox"/>	Name ▲	Type	Active since	Active till	State	Description
<input type="checkbox"/>	Server regular	With data collection	2020-04-17 00:00	2021-04-18 00:00	Active	We break and fix things at this time.

0 selected

Filter

Displaying 1 of 1 found

Dados exibidos:

Coluna	Descrição
Nome	Nome do período de manutenção. Clicar no nome do período de manutenção abre o período de manutenção formulário de configuração .
Tipo	É exibido o tipo de manutenção: Com coleta de dados ou Sem coleta de dados
Ativo desde	A data e hora em que a execução dos períodos de manutenção se torna ativa. Observação: Este horário não ativa um período de manutenção; os períodos de manutenção devem ser definidos separadamente.
Ativo até	A data e hora em que a execução dos períodos de manutenção deixa de estar ativa.
Estado	O estado do período de manutenção: Aproximando - ficará ativo em breve Ativo - está ativo Expirado - não está mais ativo
Descrição	É exibida a descrição do período de manutenção.

Para configurar um novo período de manutenção, clique no botão Criar manutenção botão período no canto superior direito.

Opções de edição em massa

Um botão abaixo da lista oferece uma opção de edição em massa:

- Excluir - exclui os períodos de manutenção

Para usar esta opção, marque as caixas de seleção antes do respectivo períodos de manutenção e clique em Excluir.

Usando filtro

Você pode usar o filtro para exibir apenas os períodos de manutenção que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de períodos de manutenção. Se você clica nele, um filtro fica disponível onde você pode filtrar períodos de manutenção por grupo de hosts, nome e estado.

Filter

Host groups

type here to search

Select

State

Any

Active

Approaching

Expired

Name

Apply

Reset

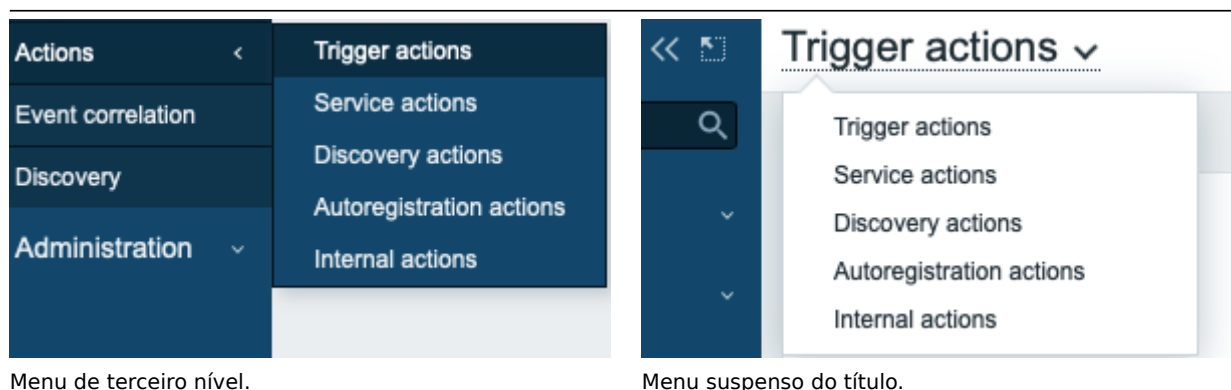
5 Ações

Visão geral

Na seção Configuração → Ações os usuários podem configurar e manter ações.

As ações exibidas são ações atribuídas à fonte de evento selecionada (gatilho, serviço, descoberta, registro automático, ações internas).

As ações são agrupadas em subseções por fonte de evento (gatilho, serviço, descoberta, registro automático, ações internas). A lista de disponíveis subseções aparece ao pressionar Ações na Configuração seção do cardápio. Também é possível alternar entre subseções usando uma lista suspensa de título no canto superior esquerdo.



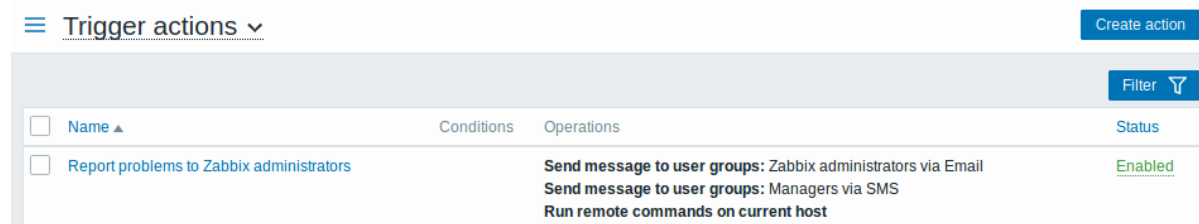
Menu de terceiro nível.

Menu suspenso do título.

Após selecionar uma subseção, uma página com uma lista de ações existentes com seus detalhes serão exibidos.

Para usuários sem direitos de superadministrador, as ações são exibidas de acordo com configurações de permissão. Isso significa que, em alguns casos, um usuário sem superadministrador direitos não é capaz de visualizar a lista de ações completa devido a certas restrições de permissão. Uma ação é exibida para o usuário sem Direitos de superadministrador se as seguintes condições forem atendidas:

- O usuário tem acesso de leitura e gravação a grupos de hosts, hosts, modelos e gatilhos em condições de ação
- O usuário tem acesso de leitura e gravação a grupos de hosts, hosts e modelos em operações de ação, operações de recuperação e operações de atualização
- O usuário tem acesso de leitura a grupos de usuários e usuários em ação operações, operações de recuperação e operações de atualização



Dados exibidos:

Coluna	Descrição
Nome	Nome da ação. Clicar no nome da ação abre a ação formulário de configuração .
Condições	As condições da ação são exibidas.
Operações	Operações de ação são exibidas. Desde o Zabbix 2.2, a lista de operações também exibe o tipo de mídia (e-mail, SMS ou script) usado para notificação, bem como o nome e sobrenome (entre parênteses após o nome de usuário) de um destinatário da notificação. A operação de ação pode ser uma notificação ou uma comando remoto dependendo do tipo de operação selecionado.
Status	O status da ação é exibido - Ativado ou Desativado. Clicando no status você pode alterá-lo. Veja os Escalations para obter mais detalhes sobre o que acontece se uma ação for desabilitada durante um escalonamento em andamento.

Para configurar uma nova ação, clique no botão Criar ação na canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

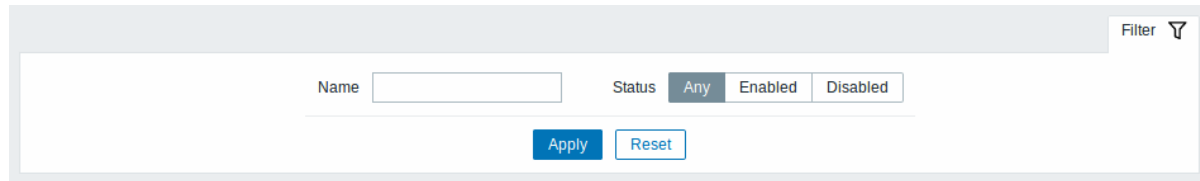
- Ativar - altere o status da ação para Ativado
- Desativar - altere o status da ação para Desativado
- Excluir - exclui as ações

Para usar essas opções, marque as caixas de seleção antes das respectivas ações, em seguida, clique no botão desejado.

Usando filtro

Você pode usar o filtro para exibir apenas as ações de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

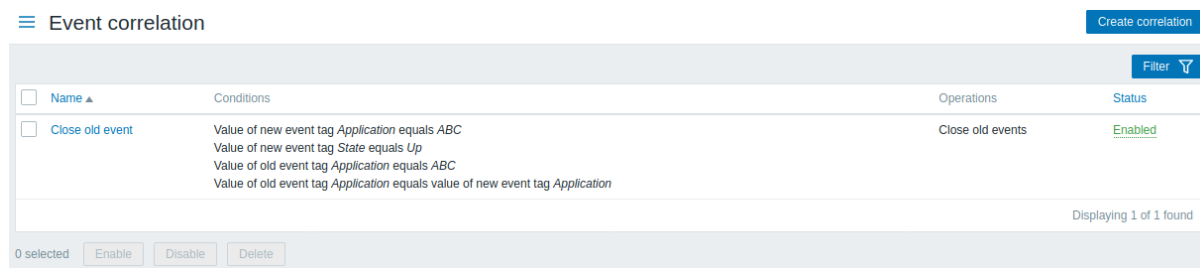
O link Filtro está disponível acima da lista de ações. Se você clicar nele, fica disponível um filtro onde você pode filtrar as ações por nome e status.



6 Correlação de eventos

Visão geral

Na seção Configuração → Correlação de eventos os usuários podem configurar e manter regras de correlação global para eventos Zabbix.



Dados exibidos:

Coluna	Descrição
Nome	Nome da regra de correlação. Clicar no nome da regra de correlação abre a regra formulário de configuração .
Condições	As condições da regra de correlação são exibidas.
Operações	As operações da regra de correlação são exibidas.
Status	O status da regra de correlação é exibido - Enabled ou Disabled. Clicando no status você pode alterá-lo.

Para configurar uma nova regra de correlação, clique no botão Criar correlação botão no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:


- Ativar - altere o status da regra de correlação para Ativado
- Desativar - altere o status da regra de correlação para Desativada
- Excluir - exclui as regras de correlação

Para usar essas opções, marque as caixas de seleção antes do respectivo regras de correlação e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas as regras de correlação que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de regras de correlação. Se você clica nele, um filtro fica disponível onde você pode filtrar regras de correlação por nome e status.

Filter 

Name
Status Any Enabled Disabled

Apply Reset


7 Descoberta

Visão geral

Na seção Configuração → Descoberta os usuários podem configurar e manter as regras de descoberta.

Uma lista de regras de descoberta existentes com seus detalhes é exibida.

Discovery rules
Create discovery rule

Filter 

<input type="checkbox"/> Name ▲	IP range	Proxy	Interval	Checks	Status
<input type="checkbox"/> Local network	192.168.0.1-254		1h	HTTP, HTTPS, SNMPv2 agent, Zabbix agent	Enabled

Displaying 1 of 1 found

0 selected
Enable
Disable
Delete

Dados exibidos:

Coluna	Descrição
Nome	Nome da regra de descoberta. Clicar no nome da regra de descoberta abre a regra de descoberta formulário de configuração .
Intervalo de IP	O intervalo de endereços IP a serem usados para varredura de rede é exibido.
Proxy	O nome do proxy é exibido, se a descoberta for realizada pelo proxy.
Intervalo	A frequência de realização da descoberta exibida.
Verificações	Os tipos de verificações usados para descoberta são exibidos.
Status	O status da ação é exibido - Enabled ou Disabled. Clicando no status você pode alterá-lo.

Para configurar uma nova regra de descoberta, clique em Criar regra de descoberta botão no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:


- Ativar - altere o status da regra de descoberta para Ativado
- Desativar - altere o status da regra de descoberta para Desativado
- Excluir - exclui as regras de descoberta

Para usar essas opções, marque as caixas de seleção antes do respectivo regras de descoberta e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas as regras de descoberta que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de regras de descoberta. Se você clique nele, um filtro fica disponível onde você pode filtrar a descoberta regras por nome e status.

Filter 

Name
Status Any Enabled Disabled

Apply Reset

6 Administração

Visão geral

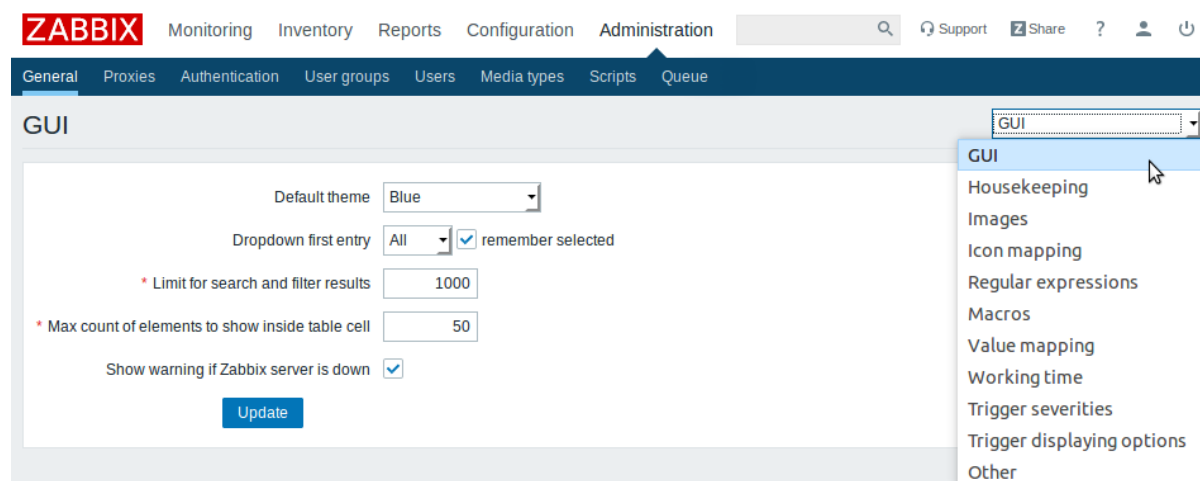
O menu Administração é para funções administrativas do Zabbix. Este menu está disponível apenas para usuários do tipo **Super Administradores**.

1 Geral

Visão geral

O módulo Administração → Geral contém várias telas de configurações relacionadas à própria interface web e customização do Zabbix.

A caixa de seleção situada no canto direito da barra de título permite alternar entre as diferentes telas de configuração.



1 GUI

Esta tela fornece customização de vários valores padrão para a interface web.

Default language	English (en_US)
Default time zone	(UTC-08:00) America/Los_Angeles
Default theme	Blue
* Limit for search and filter results	1000
* Max number of columns and rows in overview tables	50
* Max count of elements to show inside table cell	20
Show warning if Zabbix server is down	<input checked="" type="checkbox"/>
* Working time	{WORKING_HOURS}
Show technical errors	<input type="checkbox"/>
* Max history display period	24h
* Time filter default period	1h
* Max period for time selector	2y

Parâmetros de configuração:

Parâmetro	Descrição
Tema padrão	Tema a ser exibido por padrão aos usuários.
Primeira entrada da caixa suspensa	Define se a primeira entrada em uma caixa suspensa deverá ser Todos ou Nada. Em conjunto com a opção lembrar selecionado marcada, o último elemento selecionado será lembrado (ao invés do valor padrão) quando você navegar para outra página.
Limite de busca/Filtro de elementos	Quantidade máxima de elementos (linhas) que serão apresentados em uma lista na interface web, como, por exemplo, em Monitoramento → Eventos ou Configuração → Hosts. Nota: Se for definido para, por exemplo, '50', apenas os 50 primeiros elementos serão apresentados em todas as listas da interface web. Se alguma lista contiver mais de cinquenta elementos, será adicionado o sinal indicativo '+' em "Exibindo 1 a 50 de 50+ encontrados". Da mesma forma, se um filtro for utilizado e continuar existindo mais de 50 correspondências, somente os 50 primeiros serão exibidos.
Máximo de elementos mostrados\\dentro de uma célula de tabela	Para entradas que são mostradas em uma célula de uma tabela, não serão exibidos mais registros do que o aqui definido.
Ativar reconhecimento de eventos	Este parâmetro define se o reconhecimento de eventos estará ativo na interface web.
Não exibir eventos mais\\antigos que (em dias)	Este parâmetro limita qual o limite máximo (em dias) de pesquisa retroativa para a tela de status de triggers. O padrão são 7 dias.
Máximo de eventos por trigger a exibir	Quantidade máxima de eventos a serem exibidos para cada trigger na tela de status das triggers. O padrão são 100.
Exibir um alerta se o servidor Zabbix estiver fora do ar	Este parâmetro ativa a exibição de uma mensagem de alerta no navegador caso o Zabbix Server esteja fora do ar. A mensagem continua visível mesmo que a tela seja rolada para baixo. Se o mouse se posicionar em cima da mensagem ela irá desaparecer temporariamente. Suportado desde o Zabbix 2.0.1 .

2 Limpeza de dados

O processo de limpeza de dados (housekeeper) é um processo periódico executado pelo Zabbix Server. O processo remove as informações vencidas e informações apagadas pelo usuário.

Events and alerts

Enable internal housekeeping ☒

* Trigger data storage period

* Internal data storage period

* Network discovery data storage period

* Auto-registration data storage period

Services

Enable internal housekeeping ☒

* Data storage period

Audit

Enable internal housekeeping ☒

* Data storage period

User sessions

Enable internal housekeeping ☒

* Data storage period

History

Enable internal housekeeping ☒

Override item history period ☐

* Data storage period

Trends

Enable internal housekeeping ☒

Override item trend period ☐

* Data storage period

[Update](#)

[Reset defaults](#)

Nesta seção as tarefas de limpeza de dados podem ser ativadas ou desativadas em grupos:

- Eventos e alertas
- Serviços de TI
- Auditoria
- Sessões de usuário
- Histórico
- Estatísticas (médias)

Quando o processo de limpeza de dados está ativo, é possível definir o período de retenção de dados.

Para o histórico e estatísticas existem opções adicionais: Substituir o período de histórico do item e Substituir o período de histórico das estatísticas (médias). Esta opção permite que você defina de forma global por quantos dias deverão ser mantidos os dados de histórico/médias, estas opções sobrescrevem as configurações feitas diretamente nos **itens**.

Também é possível sobrescrever o período de armazenamento de histórico/médias com o processo de limpeza inativo. Assim, ao utilizar um processo externo de limpeza de dados o período de armazenamento pode ser definido usando o campo Período de armazenamento de dados (em dias).

O botão Restaurar padrão restaura as configurações desta tela ao padrão da ferramenta.

Empregada

A governanta é um processo periódico, executado pelo servidor Zabbix. O processo remove informações desatualizadas e informações excluídas pelo usuário.

Events and alerts

Enable internal housekeeping

☒

* Trigger data storage period

365d

* Service data storage period

1d

* Internal data storage period

1d

* Network discovery data storage period

1d

* Autoregistration data storage period

1d

Services

Enable internal housekeeping

☒

* Data storage period

365d

Audit

Enable internal housekeeping

☒

* Data storage period

365d

User sessions

Enable internal housekeeping

☒

* Data storage period

365d

History

Enable internal housekeeping

☒

Override item history period

☒

* Data storage period

365d

Trends

Enable internal housekeeping

☒

Override item trend period

☒

* Data storage period

365d

Update

Reset defaults

Nesta seção, as tarefas de limpeza podem ser ativadas ou desativadas em um por tarefa separadamente para: eventos e alertas/serviços de TI/usuário sessões/histórico/tendências. As configurações de manutenção de auditoria estão disponíveis em um separar [menu seção](#).

Se o serviço de limpeza estiver ativado, é possível definir por quantos dias os dados registros serão mantidos antes de serem removidos pela governanta.

A exclusão de um item/gatilho também excluirá os problemas gerados por esse item/gatilho.

Além disso, um evento só será excluído pela governanta se não for associado a um problema de alguma forma. Isso significa que, se um evento for um problema ou evento de recuperação, ele não será excluído até que o registro de problema relacionado é

removido. A governanta excluirá problemas primeiro e eventos depois, para evitar problemas potenciais com eventos obsoletos ou registros de problemas.

Para histórico e tendências, uma opção adicional está disponível: Substituir item período do histórico e Substituir período de tendência do item. Esta opção permite definido globalmente por quantos dias o histórico/tendências do item serão mantidos (1 hora a 25 anos; ou "0"), neste caso substituindo os valores definidos para itens individuais nos campos Período de armazenamento de histórico/Período de armazenamento de tendências em **configuração do item**. Observe que o período de armazenamento não será substituído por itens que possuem configuração opção Não manter histórico e/ou Não manter tendências habilitadas.

É possível substituir o período de armazenamento de histórico/tendência mesmo se a limpeza interna está desativada. Assim, ao usar uma fonte externa governanta, o período de armazenamento do histórico pode ser definido usando o histórico Campo Período de armazenamento de dados.

::: não importante Se estiver usando o TimescaleDB, para obter vantagem do particionamento automático de histórico e tendências do TimescaleDB tabelas, Substituir período do histórico do item e Substituir período de tendência do item as opções devem ser ativadas, bem como a opção Ativar limpeza interna para história e tendências. Caso contrário, os dados mantidos nessas tabelas serão ainda será armazenado em divisórias, no entanto, a governanta estará limpando o histórico e as tendências excluindo registros individuais em vez de descartando partições desatualizadas. Ao descartar partições desatualizadas é habilitado, o servidor Zabbix e o frontend não acompanharão mais os arquivos excluídos itens e histórico para itens excluídos serão apagados quando um desatualizado partição é excluída. :::

Sufixos de tempo são suportados no período campos, por exemplo 1d (um dia), 1w (uma semana). O mínimo é 1 dia (1 hora para a história), o máximo - 25 anos.

O botão Redefinir padrões permite reverter quaisquer alterações feitas.

1 Mapeamento de ícones

Esta tela permite que se definam mapeamentos entre campos do inventário dos hosts e ícones que devem ser utilizados nos mapas de rede.

Estes mapeamentos podem ser selecionados **na configuração de mapas de rede** para definir automaticamente o ícone apropriado para cada elemento do tipo 'Host'.

Para criar um novo mapeamento clique no botão Criar mapeamento de ícone localizado no canto direito da barra de título.

Parâmetros de configuração:

Parâmetro	Descrição
Nome	Identificador único do mapeamento de ícones.
Mapeamentos	Uma lista de mapeamentos. A ordem do mapeamento determina qual ícone será selecionado prioritariamente. Você pode mover os mapeamentos de posição usando o mouse.
Campo de inventário	Campo de inventário a ser consultado pela expressão.
Expressão	Expressão regular descrevendo a regra de correspondência.
Ícone	Ícone a ser utilizado caso encontre correspondência.
Padrão	Ícone padrão a ser utilizado caso o mapeamento não encontre correspondência.

2 Expressões regulares

Esta tela permite que você crie expressões regulares customizadas que poderão ser utilizadas em diversos locais da interface web. Veja mais no manual de **expressões regulares**.

3 Macros

Esta tela permite que você defina macros de usuário em nível global.

MACRO	VALUE
<code>{SSNMP_COMMUNITY}</code>	public
<code>{\$MACRO}</code>	value

[Add](#)

[Update](#)

Consulte o manual de **macros de usuário** para maiores detalhes.

4 Mapeamento de valores

Esta tela permite que você defina os mapeamentos de valores. Este recurso é particularmente útil ao traduzir códigos de resposta em representações "humanizadas" da informação.

Value mapping			Create value map	Import
<input type="checkbox"/> Name ▲	Value map	Used in items		
<input type="checkbox"/> Alarm state	0 ⇒ Ok 1 ⇒ Alarm	Yes		
<input type="checkbox"/> APC Battery Replacement Status	1 ⇒ unknown 2 ⇒ notInstalled 3 ⇒ ok 4 ⇒ failed 5 ⇒ highTemperature 6 ⇒ replaceImmediately 7 ⇒ lowCapacity			
<input type="checkbox"/> APC Battery Status	1 ⇒ unknown 2 ⇒ batteryNormal 3 ⇒ batteryLow			
<input type="checkbox"/> CIM_LogicalDevice::Availability	1 ⇒ Other 2 ⇒ Unknown 3 ⇒ Running/Full Power 4 ⇒ Warning 5 ⇒ In Test 6 ⇒ Not Applicable 7 ⇒ Power Off 8 ⇒ Off Line 9 ⇒ Off Duty 10 ⇒ Degraded 11 ⇒ Not Installed 12 ⇒ Install Error 13 ⇒ Power Save - Unknown 14 ⇒ Power Save - Low Power Mode			

Consulte o manual de **mapeamento de valores** para maiores detalhes.

5 Horário comercial

O horário comercial é um parâmetro global. Os horários comerciais irão aparecer com fundo branco em gráficos, enquanto os horários fora do horário comercial aparecerão com fundo cinza.

* Working time

Update

Consulte o manual de **definição de períodos de hora** para maiores detalhes.

6 Severidades de trigger

Esta tela permite que você customize as **severidades de trigger**, tanto seus nomes quanto cores.

Trigger severities

* Not classified	>Custom name<	97AAB3
* Information	Information	7499FF
* Warning	Warning	FFC859
* Average	Average	FFA059
* High	High	
* Disaster	Disaster	

Custom severity names affect all locales and require manual translation!

Update

Reset defaults



Você pode informar novos nomes e códigos de cor para os níveis de severidade (você também pode selecionar as cores em uma paleta de cores).

Consulte o manual de [customização de severidades de trigger](#) para maiores detalhes.

7 Opções de exibição de trigger

Esta tela permite que você configure como os status de trigger são apresentados na interface web.

Use custom event status colors ☒

* Unacknowledged PROBLEM events CC0000 ☒ blinking

* Acknowledged PROBLEM events CC0000 ☒ blinking

* Unacknowledged RESOLVED events 009900 ☒ blinking

* Acknowledged RESOLVED events 009900 ☒ blinking

* Display OK triggers for 5m

* On status change triggers blink for 2m

Update

Reset defaults

Você pode customizar as cores de eventos reconhecidos/não reconhecidos, habilitar/desabilitar a opção de piscar e modificar o tempo que deve-se dar o destaque piscante nas mudanças de estado de trigger.

8 Outros parâmetros

Esta tela permite que você configure vários outros parâmetros da interface web.

Frontend URL

Example: https://localhost/zabbix/ui/

* Group for discovered hosts

Discovered hosts x

Select

Default host inventory mode

DisabledManualAutomatic

User group for database down message

type here to search

Select

Log unmatched SNMP traps

☒

Authorization

* Login attempts

5

* Login blocking interval

30s

Security

Validate URI schemes

☒

http,https,ftp,file,mailto,tel,ssh

* Use X-Frame-Options HTTP header ?

☒

SAMEORIGIN

Use iframe sandboxing

☒

Iframe sandboxing exceptions

Communication with Zabbix server

* Network timeout

3s

* Connection timeout

3s

* Network timeout for media type test

65s

* Network timeout for script execution

60s

* Network timeout for item test

60s

* Network timeout for scheduled report test

60s

Update

Reset defaults

Parâmetro	Descrição
Atualizar itens não suportados (em segundos)	<p>Alguns itens passam para o estado 'não suportado' devido a erros em parâmetros de usuário ou por causa que o item não é suportado pelo agente. O Zabbix pode ser configurado para, de forma periódica, tentar novamente a coleta de um item 'não suportado'.</p> <p>O Zabbix irá reativar o item não suportado a cada N segundos (aqui definido). Se for definido para 0, a ativação automática não irá mais ocorrer.</p> <p>Os proxies verificam itens não suportados a cada 10 minutos, não existe possibilidade de personalização para o proxy.</p>
Grupo para hosts descobertos	<p>Os hosts descobertos por regras de descoberta de rede e pelo processo de autorregistro serão automaticamente adicionados em um grupo de host, aqui definido.</p>
Modo padrão do inventário em novos hosts	<p>Define o modo padrão de inventário para os hosts. Este valor será utilizado toda vez que um novo host ou novo protótipo de host for criado pelo Zabbix Server ou pela interface web. O valor aqui definido pode ser sobrescrito pela operação de Definir modo de inventário disponível em ações com origem em descoberta/autorregistro.</p>

Parâmetro	Descrição
Grupo de usuários que receberá a mensagem de banco de dados indisponível	<p>Grupo de usuários que receberá a notificação de problemas com o banco de dados do Zabbix.</p> <p>A disponibilidade do Zabbix Server depende da disponibilidade do seu banco de dados. Não é possível que ele funcione sem um banco de dados. O Database watchdog é um processo especial do Zabbix Server que irá alertar os usuários em caso de desastre com o banco de dados. Se o banco de dados estiver "offline", o 'watchdog' irá enviar notificação para o grupo de usuários aqui definido usando todos os tipos de mídia configurada para os usuários. O Zabbix server não irá parar; ele continuará aguardando o retorno do banco de dados para continuar o processamento.</p> <p>Nota: Antes da versão 1.8.2 o 'watchdog' era suportado apenas para MySQL. Desde a versão 1.8.2, o recurso é suportado para todos os bancos de dados.</p>
Registrar traps SNMP não correspondentes	Registrar traps SNMP que não encontrem interfaces SNMP correspondentes.

Modules

This section allows to administer custom **frontend modules**.

Modules

Scan directory

Filter

<input type="checkbox"/>	Name	Version	Author	Description	Status
<input type="checkbox"/>	Example module	1.0	John Smith	Short description of the module.	Enabled

0 selected

Enable

Disable

Displaying 1 of 1 found

Click on Scan directory to register/unregister any custom modules. Registered modules will appear in the list, along with their details. Unregistered modules will be removed from the list.

You may filter modules by name or status (enabled/disabled). Click on the module status in the list to enable/disable a module. You may also mass enable/disable modules by selecting them in the list and then clicking on the Enable/Disable buttons below the list.

API tokens

This section allows to create and manage API tokens.

API tokens

Create API token

Filter

<input type="checkbox"/>	Name	User	Expires at	Created at	Created by user	Last accessed at	Status
<input type="checkbox"/>	Token	Admin (Zabbix Administrator)	2022-01-26 00:00:00	2021-01-22 15:51:02	Admin (Zabbix Administrator)	Never	Enabled
<input type="checkbox"/>	Token 2	new_user	2021-01-26 00:00:00	2021-01-22 16:13:03	Admin (Zabbix Administrator)	Never	Enabled
<input type="checkbox"/>	Token 3	guest1	Never	2021-01-22 16:08:49	Admin (Zabbix Administrator)	Never	Enabled

You may filter API tokens by name, users to whom the tokens are assigned, expiry date, users that created tokens, or status (enabled/disabled). Click on the token status in the list to quickly enable/disable a token. You may also mass enable/disable tokens by selecting them in the list and then clicking on the Enable/Disable buttons below the list.

To create a new token, press Create API token button at the top right corner, then fill out the required fields in the token configuration screen:

API tokens ▾

* Name

Token 1

* User

Admin (Zabbix Administrator) ✕

Select

Description

Set expiration date and time

☒

* Expires at

2022-01-28 00:00:00

Enabled

☒

Add

Cancel

Parameter	Description
Name	Token's visible name.
User	User the token should be assigned to. To quickly select a user, start typing the username, first or last name, then select the required user from the auto-complete list. Alternatively, you can press the Select button and select a user from the full user list. A token can be assigned only to one user.
Description	Optional token description.
Set expiration date and time	Unmark this checkbox if a token should not have an expiry date.
Expiry date	Click on the calendar icon to select token expiry date or enter the date manually in a format YYYY-MM-DD hh:mm:ss
Enabled	Unmark this checkbox if you need to create a token in a disabled state.

Press Add to create a token. On the next screen, copy and save in a safe place Auth token value **before closing the page**, then press Close. The token will appear in the list.

Warning:

Auth token value cannot be viewed again later. It is only available immediately after creating a token. If you lose a saved token you will have to regenerate it and doing so will create a new authorization string.

Click on the token name to edit the name, description, expiry date settings, or token status. Note, that it is not possible to change to which user the token is assigned. Press Update button to save changes. If a token has been lost or exposed, you may press Regenerate button to generate new token value. A confirmation dialog box will appear, asking you to confirm this operation since after proceeding the previously generated token will become invalid.

Users without access to the Administration menu section can see and modify details of tokens assigned to them in the User profile → API tokens **section** only if Manage API tokens is allowed in their **user role** permissions.

Other parameters

This section allows configuring miscellaneous other frontend parameters.

Frontend URL

Example: https://localhost/zabbix/ui/

* Group for discovered hosts

Discovered hosts x

Select

Default host inventory mode

Disabled

Manual

Automatic

User group for database down message

type here to search

Select

Log unmatched SNMP traps

☒

Authorization

* Login attempts

5

* Login blocking interval

30s

Security

Validate URI schemes

☒

http,https,ftp,file,mailto,tel,ssh

* Use X-Frame-Options HTTP header ?

☒

SAMEORIGIN

Use iframe sandboxing

☒

Iframe sandboxing exceptions

Communication with Zabbix server

* Network timeout

3s

* Connection timeout

3s

* Network timeout for media type test

65s

* Network timeout for script execution

60s

* Network timeout for item test

60s

* Network timeout for scheduled report test

60s

Update

Reset defaults

Parameter	Description
Frontend URL	URL to Zabbix web interface. This parameter is used by Zabbix web service for communication with frontend and should be specified to enable scheduled reports.
Group for discovered hosts	Hosts discovered by network discovery and agent autoregistration will be automatically placed in the host group, selected here.
Default host inventory mode	Default mode for host inventory. It will be followed whenever a new host or host prototype is created by server or frontend unless overridden during host discovery/autoregistration by the Set host inventory mode operation.
User group for database down message	<p>User group for sending alarm message or 'None'.</p> <p>Zabbix server depends on the availability of the backend database. It cannot work without a database. If the database is down, selected users can be notified by Zabbix. Notifications will be sent to the user group set here using all configured user media entries. Zabbix server will not stop; it will wait until the database is back again to continue processing.</p> <p>Notification consists of the following content:</p> <p>[MySQL\ PostgreSQL\ Oracle] database <DB Name> [on <DB Host>:<DB Port>] is not available: <error message depending on the type of DBMS (database)></p> <p><DB Host> is not added to the message if it is defined as an empty value and <DB Port> is not added if it is the default value ("0").</p> <p>The alert manager (a special Zabbix server process) tries to establish a new connection to the database every 10 seconds. If the database is still down the alert manager repeats sending alerts, but not more often than every 15 minutes.</p>

Parameter	Description
Log unmatched SNMP traps	Log SNMP trap if no corresponding SNMP interfaces have been found.

Authorization

Parameter	Description
Login attempts	Number of unsuccessful login attempts before the possibility to log in gets blocked.
Login blocking interval	Period of time for which logging in will be prohibited when Login attempts limit is exceeded.

Security

Parameter	Description
Validate URI schemes	Uncheck the box to disable URI scheme validation against the whitelist defined in Valid URI schemes. (enabled by default).
Valid URI schemes	A comma-separated list of allowed URI schemes. Applies to all fields in the frontend where URIs are used (for example, map element URLs).
X-Frame-Options HTTP header	this field is editable only if Validate URI schemes is selected. Value of HTTP X-Frame-options header. Supported values: SAMEORIGIN (default) - the page can only be displayed in a frame on the same origin as the page itself. DENY - the page cannot be displayed in a frame, regardless of the site attempting to do so. null - disable X-Frame-options header (not recommended). Or a list (string) of comma-separated hostnames. If a listed hostname is not among allowed, the SAMEORIGIN option is used.
Use iframe sandboxing	This parameter determines whether retrieved URL content should be put into the sandbox or not. Note, that turning off sandboxing is not recommended.
Iframe sandboxing exceptions	If sandboxing is enabled and this field is empty, all sandbox attribute restrictions apply. To disable some of the restrictions, specified them in this field. This disables only restrictions listed here, other restrictions will still be applied. See sandbox attribute description for additional information.

Communication with Zabbix server

Parameter	Description
Network timeout	How many seconds to wait before closing an idle socket (if a connection to Zabbix server has been established earlier, but frontend can not finish read/send data operation during this time, the connection will be dropped). Allowed range: 1 - 300s (default: 3s).
Connection timeout	How many seconds to wait before stopping an attempt to connect to Zabbix server. Allowed range: 1 - 30s (default: 3s).
Network timeout for media type test	How many seconds to wait for a response when testing a media type. Allowed range: 1 - 300s (default: 65s).
Network timeout for script execution	How many seconds to wait for a response when executing a script. Allowed range: 1 - 300s (default: 60s).
Network timeout for item test	How many seconds to wait for returned data when testing an item. Allowed range: 1 - 300s (default: 60s).
Network timeout for scheduled report test	How many seconds to wait for returned data when testing a scheduled report. Allowed range: 1 - 300s (default: 60s).

2 Proxies

Visão geral

Na seção Administração → Proxies proxies para **distributed monitoramento** pode ser configurado no Interface Zabbix.

Proxies

Uma lista de proxies existentes com seus detalhes é exibida.

☰ Proxies

Create proxy

									Filter
<input type="checkbox"/>	Name ▾	Mode	Encryption	Compression	Last seen (age)	Host count	Item count	Required performance (vps)	Hosts
<input type="checkbox"/>	Remote proxy	Active	NONE	ON	21h 15m 15s				New host
<input type="checkbox"/>	New proxy	Active	NONE	OFF	Never				

Displaying 2 of 2 found

Dados exibidos:

Coluna	Descrição
Nome	Nome do proxy. Clicar no nome do proxy abre o proxy formulário de configuração .
Modo	Modo proxy é exibido - Ativo ou Passivo.
Criptografia	O status de criptografia para conexões do proxy é exibido: Nenhum - sem criptografia PSK - usando chave pré-compartilhada Cert - usando certificado
Última vista (idade)	É exibida a hora em que o proxy foi visto pela última vez pelo servidor.
Contagem de hosts	É exibido o número de hosts habilitados atribuídos ao proxy.
Contagem de itens	É exibido o número de itens habilitados em hosts habilitados atribuídos ao proxy.
Desempenho necessário (vps)	O desempenho de proxy necessário é exibido (o número de valores que precisam ser coletados por segundo).
Hosts	Todos os hosts monitorados pelo proxy são listados. Clicar no nome do host abre o formulário de configuração do host.

Para configurar um novo proxy, clique no botão Criar proxy na parte superior canto direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Habilitar hosts - altera o status dos hosts monitorados pelo proxy para Monitorado
- Desabilitar hosts - altera o status dos hosts monitorados pelo proxy para Não monitorado
- Excluir - exclui os proxies

Para usar essas opções, marque as caixas de seleção antes dos respectivos proxies, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os proxies de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O link Filter está disponível acima da lista de proxies. Se você clicar nele, fica disponível um filtro onde você pode filtrar proxies por nome e modo.

Filter

Name

Mode

Any

Active

Passive

Apply

Reset

3 Autenticação

[comentário]: # (tags: ldap, saml)

Visão geral

A seção Administração → Autenticação permite especificar o método de autenticação de usuário global para Zabbix e senha interna requisitos. Os métodos disponíveis são internos, HTTP, LDAP e SAML autenticação.

Autenticação padrão

Por padrão, o Zabbix usa autenticação interna do Zabbix para todos os usuários. Isto é possível alterar o método padrão para **LDAP** em todo o sistema ou ative a autenticação LDAP apenas para grupos de usuários específicos.

Para definir o LDAP como método de autenticação padrão para todos os usuários, navegue até a guia LDAP e configure os parâmetros de autenticação, depois volte para a guia Autenticação e alterne o seletor Autenticação padrão para LDAP.

Observe que o método de autenticação pode ser ajustado no **user group**. Ainda que a autenticação LDAP está definida globalmente, alguns grupos de usuários ainda podem ser autenticado pelo Zabbix. Esses grupos devem ter **frontend access** set para Interno. Vice-versa, se a autenticação interna for usada globalmente, Os detalhes de autenticação LDAP podem ser especificados e usados para usuários específicos grupos cujo **frontend access** é definido como LDAP. Se um usuário estiver incluído em pelo menos um grupo de usuários com autenticação LDAP, este usuário não poderá usar Método de autenticação.

HTTP e **SAML 2.0** métodos de autenticação podem ser usados além do padrão Método de autenticação.

LDAP

Um servidor LDAP externo pode ser utilizado para verificar os nomes e senhas dos usuários. Observe que o usuário deverá ser cadastrado no servidor LDAP e no Zabbix e que o campo de **senha** do perfil de usuário não estará mais disponível.

A autenticação LDAP do Zabbix funciona tanto com o Microsoft Active Directory quanto com o OpenLDAP.

Authentication HTTP settings **LDAP settings** SAML settings

Enable LDAP authentication ☒

* LDAP host

* Port

* Base DN

* Search attribute

Bind DN

Case sensitive login ☒

Bind password

Test authentication [must be a valid LDAP user]

* Login

* User password

Parâmetros de configuração:

Parâmetro	Descrição
Servidor LDAP	Nome do servidor LDAP. Por exemplo: ldap://ldap.zabbix.com Para conexão segura ao LDAP use o protocolo ldaps. ldaps://ldap.zabbix.com

Parâmetro	Descrição
Porta	Porta do servidor LDAP. O padrão é 389. Para o LDAP seguro a porta padrão costuma ser 636.
Base DN	Caminho base para pesquisar contas de usuário: ou=Users,ou=system (para o OpenLDAP), DC=company,DC=com (para o Microsoft Active Directory)
Atributo de pesquisa	Atributo de conta LDAP utilizado na pesquisa: uid (para o OpenLDAP), sAMAccountName (para o Microsoft Active Directory)
Bind DN	Conta LDAP para conexão e pesquisa no servidor LDAP, exemplos: uid=ldap_search,ou=system (para o OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (para o Microsoft Active Directory)
Senha para o bind	Obrigatório, conexão anônima não é suportada.
Teste de autenticação	Senha da conta de BIND no LDAP.
Login	Cabeçalho para a seção de teste Nome de um usuário de teste (que esteja atualmente conectado à interface web do Zabbix). Este nome de usuário deverá existir no servidor LDAP. O Zabbix não irá ativar a autenticação LDAP sem antes conseguir testar um usuário.
Senha do usuário	Senha do usuário de teste no LDAP.

Note:

É recomendável criar uma conta em separado para o processo de conexão (Bind DN) que possua o mínimo de privilégios possível no LDAP ao invés de utilizar uma conta real de usuário (utilizada para se autenticar na interface web do Zabbix). Esta abordagem proê mais segurança e não requer mudança na Senha para o Bind quando as contas de usuário mudarem suas próprias senhas no servidor LDAP.

Note:

Alguns grupos de usuários podem continuar sendo autenticados internamente no Zabbix. Estes grupos precisam ter o campo **acesso à interface web** definido como interno.

HTTP

A autenticação baseada em Apache (HTTP) pode ser utilizada para verificar nomes e senhas. Assim como na autenticação LDAP o usuário precisará existir no Zabbix também, e a senha do Zabbix não será utilizada.

Attention:

Atenção! Certifique-se de que a autenticação apache esteja correta e funcional antes de alterar para este modo de autenticação.

Note:

No caso da autenticação apache todos os usuário (mesmo os com o **acesso à interface web** definido como interno) serão autorizados somente pelo apache!

LDAP authentication

External LDAP authentication can be used to check user names and passwords. Note that a user must exist in Zabbix as well, however its Zabbix password will not be used.

Zabbix LDAP authentication works at least with Microsoft Active Directory and OpenLDAP.

Authentication
HTTP settings
LDAP settings
SAML settings

Enable LDAP authentication

* LDAP host

* Port

* Base DN

* Search attribute

Bind DN

Case sensitive login

Bind password

Test authentication

* Login

* User password

Configuration parameters:

Parameter	Description
Enable LDAP authentication	Mark the checkbox to enable LDAP authentication.
LDAP host	Name of LDAP server. For example: ldap://ldap.zabbix.com For secure LDAP server use ldaps protocol. ldaps://ldap.zabbix.com
Port	With OpenLDAP 2.x.x and later, a full LDAP URI of the form ldap://hostname:port or ldaps://hostname:port may be used. Port of LDAP server. Default is 389. For secure LDAP connection port number is normally 636. Not used when using full LDAP URIs.
Base DN	Base path to search accounts: ou=Users,ou=system (for OpenLDAP), DC=company,DC=com (for Microsoft Active Directory)
Search attribute	LDAP account attribute used for search: uid (for OpenLDAP), sAMAccountName (for Microsoft Active Directory)
Bind DN	LDAP account for binding and searching over the LDAP server, examples: uid=ldap_search,ou=system (for OpenLDAP), CN=ldap_search,OU=user_group,DC=company,DC=com (for Microsoft Active Directory)
Case-sensitive login	Anonymous binding is also supported. Unmark the checkbox to disable case-sensitive login (enabled by default) for usernames. E.g. disable case-sensitive login and log in with, for example, 'ADMIN' user even if the Zabbix user is 'Admin'. Note that with case-sensitive login disabled the login will be denied if multiple users exist in Zabbix database with similar usernames (e.g. Admin, admin).

Parameter	Description
Bind password	LDAP password of the account for binding and searching over the LDAP server.
Test authentication	Header of a section for testing
Login	Name of a test user (which is currently logged in the Zabbix frontend). This user name must exist in the LDAP server. Zabbix will not activate LDAP authentication if it is unable to authenticate the test user.
User password	LDAP password of the test user.

Warning:

In case of trouble with certificates, to make a secure LDAP connection (ldaps) work you may need to add a `TLS_REQCERT allow` line to the `/etc/openldap/ldap.conf` configuration file. It may decrease the security of connection to the LDAP catalog.

Note:

It is recommended to create a separate LDAP account (Bind DN) to perform binding and searching over the LDAP server with minimal privileges in the LDAP instead of using real user accounts (used for logging in the Zabbix frontend). Such an approach provides more security and does not require changing the Bind password when the user changes his own password in the LDAP server. In the table above it's `ldap_search` account name.

SAML authentication

SAML 2.0 authentication can be used to sign in to Zabbix. Note that a user must exist in Zabbix, however, its Zabbix password will not be used. If authentication is successful, then Zabbix will match a local username with the username attribute returned by SAML.

Note:

If SAML authentication is enabled, users will be able to choose between logging in locally or via SAML Single Sign-On.

Setting up the identity provider

In order to work with Zabbix, a SAML identity provider (onelogin.com, auth0.com, okta.com, etc.) needs to be configured in the following way:

- Assertion Consumer URL should be set to `<path_to_zabbix_ui>/index_sso.php?acs`
- Single Logout URL should be set to `<path_to_zabbix_ui>/index_sso.php?sls`

`<path_to_zabbix_ui>` examples: `%% https://example.com/zabbix/ui, http://another.example.com/zabbix, http://<any_public_ip_address> %%`

Setting up Zabbix

Attention:

It is required to install `php-openssl` if you want to use SAML authentication in the frontend.

To use SAML authentication Zabbix should be configured in the following way:

1. Private key and certificate should be stored in the `ui/conf/certs/`, unless custom paths are provided in `zabbix.conf.php`.

By default, Zabbix will look in the following locations:

- `ui/conf/certs/sp.key` - SP private key file
- `ui/conf/certs/sp.crt` - SP cert file
- `ui/conf/certs/idp.crt` - IDP cert file

2. All of the most important settings can be configured in the Zabbix frontend. However, it is possible to specify additional settings in the `configuration file`.

Authentication
HTTP settings
LDAP settings
SAML settings

Enable SAML authentication
☒

* IdP entity ID
https://webauth.airport.com/idp

* SSO service URL
https://idp.airport.com/idp/profile/SAML2/SSO/f76245e

SLO service URL

* Username attribute
uid

* SP entity ID
https://intranet.jfk.airport.com/sp

SP name ID format
urn:oasis:names:tc:SAML:2.0:nameid-format:transient

Sign
☐ Messages
☐ Assertions
☒ AuthN requests
☐ Logout requests
☐ Logout responses

Encrypt
☐ Name ID
☐ Assertions

Case sensitive login
☒

Configuration parameters, available in the Zabbix frontend:

Parameter	Description
Enable SAML authentication	Mark the checkbox to enable SAML authentication.
IDP entity ID	The unique identifier of SAML identity provider.
SSO service URL	The URL users will be redirected to when logging in.
SLO Service URL	The URL users will be redirected to when logging out. If left empty, the SLO service will not be used.
// Username attribute//	SAML attribute to be used as a username when logging into Zabbix. List of supported values is determined by the identity provider. Examples: uid userprincipalname samaccountname username userusername urn:oid:0.9.2342.19200300.100.1.1 urn:oid:1.3.6.1.4.1.5923.1.1.1.13 urn:oid:0.9.2342.19200300.100.1.44
SP entity ID	The unique identifier of SAML service provider.

Parameter	Description
SP name ID format	Defines which name identifier format should be used. Examples: urn:oasis:names:tc:SAML:2.0:nameid-format:persistent urn:oasis:names:tc:SAML:2.0:nameid-format:transient urn:oasis:names:tc:SAML:2.0:nameid-format:kerberos urn:oasis:names:tc:SAML:2.0:nameid-format:entity
Sign	Mark the checkboxes to select entities for which SAML signature should be enabled: Messages Assertions AuthN requests Logout requests Logout responses
Encrypt	Mark the checkboxes to select entities for which SAML encryption should be enabled: Assertions Name ID
Case-sensitive login	Mark the checkbox to enable case-sensitive login (disabled by default) for usernames. E.g. disable case-sensitive login and log in with, for example, 'ADMIN' user even if the Zabbix user is 'Admin'. Note that with case-sensitive login disabled the login will be denied if multiple users exist in Zabbix database with similar usernames (e.g. Admin, admin).

Advanced settings

Additional SAML parameters can be configured in the Zabbix frontend configuration file (zabbix.conf.php):

- `$SSO['SP_KEY'] = '<path to the SP private key file>';`
- `$SSO['SP_CERT'] = '<path to the SP cert file>';`
- `$SSO['IDP_CERT'] = '<path to the IDP cert file>';`
- `$SSO['SETTINGS']`

Note:

Zabbix uses [OneLogin's SAML PHP Toolkit](#) library (version 3.4.1). The structure of `$SSO['SETTINGS']` section should be similar to the structure used by the library. For the description of configuration options, see official library [documentation](#).

Only the following options can be set as part of `$SSO['SETTINGS']`:

- `strict`
- `baseurl`
- `compress`
- `contactPerson`
- `organization`
- `sp` (only options specified in this list)
 - `attributeConsumingService`
 - `x509certNew`
- `idp` (only options specified in this list)
 - `singleLogoutService` (only one option)
 - * `responseUrl`
 - `certFingerprint`
 - `certFingerprintAlgorithm`
 - `x509certMulti`
- `security` (only options specified in this list)
 - `signMetadata`
 - `wantNameId`
 - `requestedAuthnContext`
 - `requestedAuthnContextComparison`
 - `wantXMLValidation`
 - `relaxDestinationValidation`

- destinationStrictlyMatches
- rejectUnsolicitedResponsesWithInResponseTo
- signatureAlgorithm
- digestAlgorithm
- lowercaseUrlencoding

All other options will be taken from the database and cannot be overridden. The debug option will be ignored.

In addition, if Zabbix UI is behind a proxy or a load balancer, the custom use_proxy_headers option can be used:

- false (default) - ignore the option;
- true - use X-Forwarded-* HTTP headers for building the base URL.

Configuration example:

```
$SSO['SETTINGS'] = [
    'security' => [
        'signatureAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#rsa-sha384'
        'digestAlgorithm' => 'http://www.w3.org/2001/04/xmldsig-more#sha384',
        // ...
    ],
    // ...
];
```

4 Grupos de usuários

Visão geral

Na seção Administração → Grupos de usuários grupos de usuários do sistema são mantidos.

Grupos de usuários

Uma lista de grupos de usuários existentes com seus detalhes é exibida.

≡

User groups

Create user group

<input type="checkbox"/>	Name ▲	#	Members	Frontend access	Debug mode	Status
<input type="checkbox"/>	Disabled	Users 1	guest	System default	Disabled	Disabled
<input type="checkbox"/>	Enabled debug mode	Users		System default	Enabled	Enabled
<input type="checkbox"/>	Guests	Users 1	guest	Internal	Disabled	Enabled
<input type="checkbox"/>	No access to the frontend	Users		Disabled	Disabled	Enabled
<input type="checkbox"/>	Zabbix administrators	Users 1	Admin (Zabbix Administrator)	System default	Disabled	Enabled

0 selected

Enable

Disable

Enable debug mode

Disable debug mode

Delete

Filter

Displaying 5 of 5 found

Dados exibidos:

Coluna	Descrição
Nome	Nome do grupo de usuários. Clicar no nome do grupo de usuários abre o grupo de usuários formulário de configuração .
#	O número de usuários no grupo. Clicar em Usuários exibirá os respectivos usuários filtrados na lista de usuários.
Membros	Nomes de usuários individuais no grupo de usuários (com nome e sobrenome entre parênteses). Clicar no nome de usuário abrirá o formulário de configuração do usuário. Os usuários de grupos desabilitados são exibidos em vermelho.
Acesso de frontend	Nível de acesso de frontend é exibido: Padrão do sistema - Autenticação Zabbix, LDAP ou HTTP; dependendo do [método] de autenticação escolhido (autenticação) Interno - o usuário é autenticado pelo Zabbix independentemente das configurações do sistema Desativado - o acesso de frontend para este usuário está desabilitado. Ao clicar no nível atual, você pode alterá-lo.
Modo de depuração	O status do modo de depuração é exibido - Ativado ou Desativado. Ao clicar no status você pode alterá-lo.

Coluna	Descrição
Status	O status do grupo de usuários é exibido - Ativado ou Desativado. Ao clicar no status você pode alterá-lo.

Para configurar um novo grupo de usuários, clique no botão Criar grupo de usuários no canto superior direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altera o status do grupo de usuários para Ativado
- Desativar - altera o status do grupo de usuários para Desativado
- Ativar modo de depuração - ativar o modo de depuração para os grupos de usuários
- Desabilitar o modo de depuração - desabilite o modo de depuração para os grupos de usuários
- Excluir - exclui os grupos de usuários

Para usar essas opções, marque as caixas de seleção antes do respectivo usuário grupos e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os grupos de usuários que você está interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de grupos de usuários. Se você clique nele, um filtro fica disponível onde você pode filtrar grupos de usuários por nome e status.

5 Funções do usuário

Visão geral

Na seção Administração → Funções do usuário funções que podem ser atribuídas aos usuários do sistema e as permissões específicas para cada função são mantidas.

Funções de usuário padrão

Por padrão, o Zabbix é configurado com quatro funções de usuário, que possuem um conjunto predefinido de permissões:

- Função de administrador
- Função de convidado
- Função de superadministrador
- Papel do usuário

<input type="checkbox"/> Name ▲	#	Users
<input type="checkbox"/> Admin role	Users 1	db_manager (Database manager)
<input type="checkbox"/> Guest role	Users	
<input type="checkbox"/> Super admin role	Users 2	Admin (Zabbix Administrator), ljohnson (Lewis Johnson)
<input type="checkbox"/> User role	Users 4	gsione (George Slone), guest (John Snow), test_admin, test_guest

Note:

A função de superadministrador padrão não pode ser modificada ou excluída, porque pelo menos um usuário superadministrador com privilégios ilimitados deve existir no Zabbix.

Usuários do Zabbix com tipo Super admins e permissões apropriadas podem modificar ou excluir funções existentes ou criar novas funções personalizadas.

Para criar uma nova função, clique no botão Criar função de usuário na parte superior Esquina direita. Para atualizar uma função existente, pressione o nome da função para abrir o formulário de configuração.

* Name

User type

Access to UI elements

Monitoring	<input checked="" type="checkbox"/> Dashboard	<input checked="" type="checkbox"/> Hosts	<input checked="" type="checkbox"/> Maps
	<input checked="" type="checkbox"/> Problems	<input checked="" type="checkbox"/> Latest data	<input checked="" type="checkbox"/> Discovery
Services	<input checked="" type="checkbox"/> Services	<input checked="" type="checkbox"/> SLA	
	<input checked="" type="checkbox"/> Service actions	<input checked="" type="checkbox"/> SLA report	
Inventory	<input checked="" type="checkbox"/> Overview	<input checked="" type="checkbox"/> Hosts	
Reports	<input type="checkbox"/> System information	<input checked="" type="checkbox"/> Triggers top 100	<input checked="" type="checkbox"/> Notifications
	<input checked="" type="checkbox"/> Scheduled reports	<input type="checkbox"/> Audit	
	<input checked="" type="checkbox"/> Availability report	<input type="checkbox"/> Action log	
Configuration	<input checked="" type="checkbox"/> Host groups	<input checked="" type="checkbox"/> Maintenance	<input checked="" type="checkbox"/> Discovery
	<input checked="" type="checkbox"/> Templates	<input checked="" type="checkbox"/> Actions	
	<input checked="" type="checkbox"/> Hosts	<input type="checkbox"/> Event correlation	
Administration	<input type="checkbox"/> General	<input type="checkbox"/> User groups	<input type="checkbox"/> Media types
	<input type="checkbox"/> Proxies	<input type="checkbox"/> User roles	<input type="checkbox"/> Scripts
	<input type="checkbox"/> Authentication	<input type="checkbox"/> Users	<input type="checkbox"/> Queue

* At least one UI element must be checked.

Opções de permissão disponíveis junto com conjuntos de permissões padrão para funções de usuário pré-existent no Zabbix são descritas abaixo.

Parâmetro	Descrição	Funções de usuário padrão		
		Função de superadministrador	Função de administrador	Função de usuário comum
Nome	Nome visível da função.	Função de superadministrador	Função de administrador	Função de usuário comum

Parâmetro	Descrição	Funções de usuário padrão		
Tipo de usuário	O tipo de usuário selecionado determina a lista de permissões disponíveis. Ao selecionar um tipo de usuário, todas as permissões disponíveis para esse tipo de usuário são concedidas por padrão. Desmarque a(s) caixa(s) de seleção para revogar determinadas permissões para o função de usuário. As caixas de seleção para permissões não disponíveis para este tipo de usuário estão esmaecidas.	Superadministrador	Administrador	Usuário
Acesso aos elementos da interface do usuário				
Monitoramento				
Painel	Ativar/desativar o acesso a uma seção específica do menu Monitoramento e páginas subjacentes.	Sim	Sim	Sim
Problemas				
Anfitriões				
Últimos dados				
Mapas				
Descoberta			Não	Não
Serviços				
Serviços	Ativar/desativar o acesso a uma seção específica do menu Serviços e páginas subjacentes.	Sim	Sim	Sim
Ações de serviço				
SLA				
Relatório de SLA				
Inventário				
Visão geral	Ativar/desativar o acesso a uma seção específica do menu Inventário e páginas subjacentes.	Sim	Sim	Sim
Anfitriões				
Relatórios				
Informações do sistema	Ativar/desativar o acesso a uma seção específica do menu Relatórios e páginas subjacentes.	Sim	Não	Não
Relatório de disponibilidade			Sim	Sim
Aciona os 100 primeiros				
Auditoria			Não	Não
Registro de ações				
Notificações			Sim	
Relatórios programados				
Configuração				
Grupos de hosts	Ativar/desativar o acesso a uma seção específica do menu Configuração e páginas subjacentes.	Sim	Sim	Não

Parâmetro	Descrição	Funções de usuário padrão		
Modelos				
Anfitriões				
Manutenção				
Ações				
Correlação de eventos			Não	
Descoberta			Sim	
Administração				
Geral	Ativar/desativar o acesso a uma seção específica do menu Administração e páginas subjacentes.	Sim	Não	Não Não
Proxies				
Autenticação				
Grupos de usuários				
Funções do usuário				
Usuários				
Tipos de mídia				
Scripts				
Fila				
Acesso padrão aos novos elementos da interface do usuário	Ativar/desativar o acesso aos elementos personalizados da interface do usuário. Módulos, se presentes, serão listados abaixo.	Sim	Sim	Sim Sim
Acesso aos serviços				
Acesso de leitura e gravação aos serviços	<p>Selecione o acesso de leitura e gravação aos serviços:</p> <p>Nenhum - nenhum acesso</p> <p>Todos - o acesso a todos os serviços é de leitura e gravação</p> <p>* *Lista de serviços** - selecione serviços para acesso de leitura e gravação</p>	Sim	Sim	Não Não
Acesso de leitura-gravação aos serviços com tag	<p>O acesso de leitura e gravação tem precedência sobre o acesso somente leitura e é herdado dinamicamente pelos serviços filho.</p> <p>Especifique o nome da tag e, opcionalmente, o valor para conceder acesso de leitura e gravação aos serviços que correspondem à tag.</p> <p>Esta opção está disponível se 'Lista de serviços' estiver selecionada no Read-acesso de gravação ao parâmetro services.</p> <p>O acesso de leitura e gravação tem precedência sobre o acesso somente leitura e é herdado dinamicamente pelos serviços filho.</p>			

Parâmetro	Descrição	Funções de usuário padrão		
Acesso somente leitura aos serviços	<p>Selecione o acesso somente leitura aos serviços:</p> <p>Nenhum - nenhum acesso</p> <p>Todos - o acesso a todos os serviços é somente leitura</p> <p>* *Lista de serviços** - selecione serviços para acesso somente leitura</p> <p>O acesso somente leitura não tem precedência sobre o acesso leitura-gravação e é herdado dinamicamente pelos serviços filho.</p>			
Acesso somente leitura aos serviços com tag	<p>Especifique o nome da tag e, opcionalmente, o valor para conceder adicionalmente acesso somente leitura aos serviços que correspondem à tag. Esta opção está disponível se 'Lista de serviços' estiver selecionada no Read-parâmetro de acesso somente aos serviços. O acesso somente leitura não tem precedência sobre o acesso leitura-gravação e é herdado dinamicamente pelos serviços filho.</p>			
Acesso aos módulos <Nome do módulo>	<p>Permitir/negar acesso a um módulo específico. Somente os módulos habilitados são mostrados nesta seção. Não é possível conceder ou restringir o acesso a um módulo atualmente desabilitado.</p>	Sim	Sim	Sim
Acesso padrão a novos módulos	<p>Ativar/desativar o acesso aos módulos que podem ser adicionado no futuro.</p>			
Acesso à API Ativado	<p>Ativar/desativar o acesso à API.</p>	Sim	Sim	Não

Parâmetro	Descrição	Funções de usuário padrão		
Métodos de API	<p>Selecione Lista de permissões para permitir apenas métodos de API especificados ou Lista de negações para restringir apenas métodos de API especificados.</p> <p>No campo de pesquisa, comece a digitar o nome do método e selecione o método da lista de preenchimento automático. Você também pode pressionar o botão Selecionar e selecionar métodos da lista completa disponível para esse tipo de usuário. Observe que, se determinada ação do bloco Acesso a ações estiver desmarcada, os usuários não poderão usar métodos de API relacionados a essa ação.</p> <p>Os caracteres curinga são compatíveis. Exemplos: <code>dashboard.*</code> (todos os métodos do serviço de API 'dashboard.') * (qualquer método), <code>*.export</code> (métodos com nome '.export' de todos os serviços de API).</p> <p>
Se nenhum método tiver sido especificado, a regra da lista Permitir/Negar será ignorada.</p>			
Acesso às ações				
Criar e editar painéis	Desmarcar esta caixa de seleção também revogará os direitos de usar os métodos de API <code>.create</code> , <code>.update</code> e <code>.delete</code> para os elementos correspondentes.	Sim	Sim	Sim Não
Criar e editar mapas				
Criar e editar manutenção				Não
Adicionar comentários do problema	Desmarcar esta caixa de seleção também revogará os direitos de executar a ação correspondente por meio do método API <code>event.acknowledge</code> .			Yes
Alterar gravidade				
Reconhecer problemas				
Fechar problemas				

Parâmetro	Descrição	Funções de usuário padrão
Execute scripts	Desmarcar esta caixa de seleção também revogará os direitos de uso do método API <code>script.execute</code> .	
Gerenciar tokens de API	Desmarcar esta caixa de seleção também revogará os direitos de usar todos os métodos de API <code>token..</code>	
Gerenciar relatórios programados	Desmarcar esta caixa de seleção também revogará os direitos de usar todos os métodos de API <code>report..</code>	Não
Gerenciar SLA	Ativar/desativar os direitos para gerenciar SLA .	
Acesso padrão a novas ações	Ativar/desativar acesso a novas ações.	Sim

Notas:

- Cada usuário pode ter apenas uma função atribuída.
- Se um elemento for restrito, os usuários não poderão acessá-lo mesmo inserindo uma URL direta para este elemento no navegador.
- Usuários do tipo Usuário ou Admin não podem alterar sua própria função definições.
- Usuários do tipo Super admin podem modificar as configurações de sua própria função (não disponível para a função padrão de Superadministrador), mas não para o usuário tipo.
- Usuários de todos os níveis não podem alterar seu próprio tipo de usuário.

Veja também:

- [Configurando um usuário](#)

6 Usuários

Visão geral

Na seção Administração → Usuários os usuários do sistema são mantido.

Comercial

Uma lista de usuários existentes com seus detalhes é exibida.

≡

Users

User group

All

▼

Create user

<input type="checkbox"/>	Username ▲	Name	Last name	User role	Groups	Is online?	Login	Frontend access	API access	Debug mode	Status
<input type="checkbox"/>	Admin	Zabbix	Administrator	Super admin role	Zabbix administrators	Yes (03/02/2021 02:35:19 PM)	Ok	System default	Enabled	Disabled	Enabled
<input type="checkbox"/>	Database manager	James	Hughes	Admin role	DB administrators	No	Ok	System default	Enabled	Disabled	Enabled
<input type="checkbox"/>	guest			User role	Disabled, Guests	No	Ok	Internal	Enabled	Disabled	Disabled

0 selected

Unblock

Delete

Displaying 3 of 3 found

No menu suspenso à direita na barra Usuários, você pode escolher se para exibir todos os usuários ou aqueles pertencentes a um determinado grupo.

Dados exibidos:

Coluna	Descrição
Nome de usuário	Nome de usuário para logar no Zabbix. Clicar no nome de usuário abre o usuário formulário de configuração .
Nome	Nome do usuário.
Sobrenome	Segundo nome do usuário.
Função do usuário	Função do usuário é exibido.

Coluna	Descrição
Grupos	Grupos dos quais o usuário é membro são listados. Clicar no nome do grupo de usuários abre o formulário de configuração do grupo de usuários. Grupos desabilitados são exibidos em vermelho.
Está online?	O status online do usuário é exibido - Sim ou Não. A hora da última atividade do usuário é exibida entre parênteses.
Login	O status de login do usuário é exibido - Ok ou Bloqueado. Um usuário pode ficar temporariamente bloqueado ao exceder o número de tentativas de login malsucedidas definidas na seção Administration→General (cinco por padrão). Ao clicar em Bloqueado você pode desbloquear o usuário.
Acesso de frontend	O nível de acesso de frontend é exibido - Padrão do sistema, Interno ou Desativado, dependendo do definido para todo o grupo de usuários.
Acesso à API	O status de acesso à API é exibido - Ativado ou Desativado, dependendo do definido para a função do usuário.
Modo de depuração	O status do modo de depuração é exibido - Ativado ou Desativado, dependendo do definido para todo o grupo de usuários.
Status	O status do usuário é exibido - Ativado ou Desativado, dependendo do definido para todo o grupo de usuários.

Para configurar um novo usuário, clique no botão Criar usuário na parte superior canto direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Desbloquear - reativar o acesso ao sistema para usuários bloqueados
- Excluir - excluir os usuários

Para usar essas opções, marque as caixas de seleção antes dos respectivos usuários, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os usuários nos quais está interessado. Para melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de usuários. Se você clicar em ele, um filtro fica disponível onde você pode filtrar usuários por nome de usuário, nome, sobrenome e função de usuário.

7 Tipos de mídia

Visão geral

Na seção Administração → Tipos de mídia os usuários podem configurar e manter as informações do tipo de mídia.

As informações do tipo de mídia contêm instruções gerais para usar uma mídia como canal de entrega de notificações. Detalhes específicos, como a endereços de e-mail individuais para enviar uma notificação são mantidos com usuários individuais.

Uma lista de tipos de mídia existentes com seus detalhes é exibida.

<input type="checkbox"/>	Name ▲	Type	Status	Used in actions	Details	Action
<input type="checkbox"/>	Email	Email	Enabled		SMTP server: "mail.zabbix.com", SMTP helo: "zabbix.com", SMTP email: "zabbix-info@zabbix.com"	Test
<input type="checkbox"/>	Email (HTML)	Email	Enabled		SMTP server: "mail.example.com", SMTP helo: "example.com", SMTP email: "zabbix@example.com"	Test
<input type="checkbox"/>	Mattermost	Webhook	Enabled			Test
<input type="checkbox"/>	Notification script	Script	Enabled		Script name: "notification.sh"	Test
<input type="checkbox"/>	Opsgenie	Webhook	Enabled			Test
<input type="checkbox"/>	PagerDuty	Webhook	Enabled			Test
<input type="checkbox"/>	Pushover	Webhook	Enabled			Test
<input type="checkbox"/>	SMS	SMS	Enabled		GSM modem: "/dev/ttyS0"	Test

Displaying 8 of 8 found

0 selected [Enable](#) [Disable](#) [Export](#) [Delete](#)

Dados exibidos:

Coluna	Descrição
Nome	Nome do tipo de mídia. Clicar no nome abre o tipo de mídia formulário de configuração .
Tipo	O tipo de mídia (e-mail, SMS, etc) é exibido.
Status	O status do tipo de mídia é exibido - Enabled ou Disabled. Clicando no status você pode alterá-lo.
Usado em ações	Todas as ações em que o tipo de mídia é usado diretamente (selecionado na lista suspensa Enviar apenas para) são exibidas. Clicar no nome da ação abre o formulário de configuração da ação.
Detalhes	Informações detalhadas do tipo de mídia são exibidas.
Ações	A seguinte ação está disponível: Test - clique para abrir um formulário de teste onde você pode inserir parâmetros de tipo de mídia (por exemplo, um endereço de destinatário com assunto e corpo de teste) e enviar uma mensagem de teste para verifique se o tipo de mídia configurado funciona. Veja também: Teste de tipo de mídia .

Para configurar um novo tipo de mídia, clique no botão Criar tipo de mídia no canto superior direito.

Para importar um tipo de mídia de XML, clique no botão Importar na parte superior canto direito.

Opções de edição em massa

Os botões abaixo da lista oferecem algumas opções de edição em massa:

- Ativar - altere o status do tipo de mídia para Ativado
- Desativar - altere o status do tipo de mídia para Desativado
- Exportar - exporta os tipos de mídia para um arquivo YAML, XML ou JSON
- Excluir - exclua os tipos de mídia

Para usar essas opções, marque as caixas de seleção antes da respectiva mídia tipos e, em seguida, clique no botão necessário.

Usando filtro

Você pode usar o filtro para exibir apenas os tipos de mídia que você interesse. Para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvidas.

O link Filtro está disponível acima da lista de tipos de mídia. Se você clique nele, um filtro fica disponível onde você pode filtrar os tipos de mídia por nome e status.

Filter

Name

Status

Any

Enabled

Disabled

Apply

Reset

8 Scripts

Visão geral

Na seção Administração → Scripts scripts globais definidos pelo usuário pode ser configurado e mantido.

Scripts globais, dependendo do escopo configurado e também de permissões, estão disponíveis para execução:

- do **host menu** em vários locais de frontend (Dashboard, Problems, Latest dados, Mapas, etc.)
- do **evento menu**
- pode ser executado como uma operação de ação

Os scripts são executados no agente Zabbix, servidor Zabbix (proxy) ou Apenas servidor Zabbix. Veja também **Comando execução**.

Tanto no agente Zabbix quanto nos scripts remotos do proxy Zabbix são desabilitados por predefinição. Eles podem ser habilitados por:

- adicionando o parâmetro `AllowKey=system.run[*]` no agente configuração;
- definindo o parâmetro `EnableRemoteCommands` para '1' no proxy configuração

Uma lista de scripts existentes com seus detalhes é exibida.

Scripts

Create script

<input type="checkbox"/> Name ▾	Scope	Used in actions	Type	Execute on	Commands	User group	Host group	Host access
<input type="checkbox"/> Traceroute	Manual host action		Script	Server (proxy)	/usr/bin/traceroute {HOST.CONN}	All	All	Read
<input type="checkbox"/> Restart webserver	Action operation		Script	Agent	sudo /etc/init.d/apache2 restart	All	All	Read
<input type="checkbox"/> Detect operating system	Manual host action		Script	Server (proxy)	sudo /usr/bin/nmap -O {HOST.CONN}	Zabbix administrators	All	Read

Displaying 3 of 3 found

Dados exibidos:

Coluna	Descrição
Nome	Nome do script. Clicar no nome do script abre o script formulário de configuração .
Escopo	Escopo do script - operação de ação, ação manual do host ou ação manual do evento. Essa configuração determina onde o script está disponível.
Usado em ações	Ações onde o script é usado são exibidas.
Tipo	O tipo de script é exibido - comando Webhook, Script, SSH, Telnet ou IPMI.
Execute on	É exibido se o script será executado apenas no agente Zabbix, servidor Zabbix (proxy) ou servidor Zabbix apenas.
Comandos	Todos os comandos a serem executados dentro do script são exibidos.
Grupo de usuários	O grupo de usuários para o qual o script está disponível é exibido (ou Todos para todos os grupos de usuários).
Grupo de hosts	O grupo de hosts para o qual o script está disponível é exibido (ou All para todos os grupos de hosts).
Acesso ao host	O nível de permissão para o grupo de hosts é exibido - Read ou Write. Somente usuários com o nível de permissão necessário terão acesso à execução do script.

Para configurar um novo script, clique no botão Criar script na canto superior direito.

Opções de edição em massa

Um botão abaixo da lista oferece uma opção de edição em massa:


- Excluir - exclui os scripts

Para usar esta opção, marque as caixas de seleção antes dos respectivos scripts e clique em Excluir.

Usando filtro

Você pode usar o filtro para exibir apenas os scripts de seu interesse para um melhor desempenho de pesquisa, os dados são pesquisados com macros não resolvido.

O link Filter está disponível acima da lista de scripts. Se você clicar nele, fica disponível um filtro onde você pode filtrar scripts por nome e escopo.

Filter 

Name
Scope Any Action operation Manual host action Manual event action

Apply Reset

Configurando um script global

* Name

Scope

Action operation Manual host action Manual event action

Menu path

Type

Webhook Script SSH Telnet IPMI

Execute on

Zabbix agent Zabbix server (proxy) Zabbix server

* Commands

Description

Host group

User group

Required host permissions

Read Write

Enable confirmation

☐

Confirmation text

Add Cancel

Atributos do script:

Parâmetro	Descrição
Nome	Nome exclusivo do script. Ex. Limpar sistema de arquivos /tmp

Parâmetro	Descrição
Escopo	<p>Escopo do script - operação de ação, ação manual do host ou ação manual do evento. Esta configuração determina onde o script pode ser usado - em comandos remotos de operações de ação, a partir do menu do host ou do menu do evento respectivamente.</p> <p>Definir o escopo para 'Operação de ação' torna o script disponível para todos os usuários com acesso a Configuração → Ações.</p> <p>Se um script for realmente usado em um ação, seu escopo não pode ser alterado de 'operação de ação'.</p> <p>Suporte a macro</p> <p>O escopo afeta o intervalo de macros disponíveis. Por exemplo, macros relacionadas ao usuário ({USER.*}) são suportadas em scripts para permitir a transmissão de informações sobre o usuário que iniciou o script. No entanto, eles não são compatíveis se o escopo do script for uma operação de ação, pois as operações de ação são executadas automaticamente.</p> <p>Para descobrir quais macros são compatíveis, faça uma pesquisa por 'Notificações e comandos baseados em acionador/Comandos baseados em acionador', 'Scripts de ação de host manual' e 'Scripts de ação de evento manual' na tabela macros com suporte. Observe que, se uma macro pode resolver um valor com espaços (por exemplo, nome do host), não se esqueça de citar conforme necessário.</p>
Caminho do menu	<p>O caminho do menu desejado para o script. Por exemplo, Default ou Default/, exibirá o script no respectivo diretório. Os menus podem ser aninhados, por exemplo. Menu principal/Submenu1/Submenu2. Ao acessar scripts através do menu host/evento nas seções de monitoramento, eles serão organizados de acordo com os diretórios fornecidos.</p> <p>Este campo é exibido apenas se 'Ação manual do host' ou 'Ação manual do evento' estiver selecionado como Escopo.</p>
Type	<p>Clique no respectivo botão para selecionar o tipo de script:</p> <p>Webhook, Script, SSH, Telnet ou IPMI comando.</p>
Tipo de script: Webhook Parâmetros	<p>Especifique as variáveis do webhook como pares atributo-valor.</p> <p>Veja também: Webhook configuração de mídia. Macros e macros de usuário personalizados são suportados em valores de parâmetro. O suporte a macro depende do escopo do script (consulte Escopo acima).</p>
Script	<p>Insira o código JavaScript no bloco que aparece ao clicar no campo de parâmetro (ou no botão visualizar/editar próximo a ele).</p> <p>O suporte a macro depende do escopo do script (consulte Escopo acima).</p> <p>Veja também: configuração de mídia Webhook, objetos Javascript adicionais.</p>

Parâmetro	Descrição
Timeout	Tempo limite de execução JavaScript (1-60s, padrão 30s). Sufixos de tempo são suportados, por exemplo, 30s, 1m.
Tipo de script: script Execute on	Clique no respectivo botão para executar o shell script em: Agente Zabbix - o script será executado pelo agente Zabbix (se o item system.run for permitido) no host servidor Zabbix (proxy) - o script será executado pelo servidor Zabbix ou proxy (se habilitado por EnableRemoteCommands) - dependendo se o host é monitorado por servidor ou proxy Zabbix server - o script será executado apenas pelo Zabbix server
Comandos	Digite o caminho completo para os comandos a serem executados dentro do script. O suporte a macro depende do escopo do script (veja Escopo acima). macros de usuário personalizados são suportados.
Tipo de script: SSH Método de autenticação Nome de usuário Senha	Selecione o método de autenticação - senha ou chave pública. Digite o nome de usuário.
Arquivo de chave pública	Digite a senha. Este campo estará disponível se 'Senha' for selecionado como método de autenticação.
Arquivo de chave privada	Digite o caminho para o arquivo de chave pública. Este campo está disponível se 'Chave pública' for selecionado como método de autenticação.
Passphrase	Digite o caminho para o arquivo de chave privada. Este campo está disponível se 'Chave pública' for selecionado como método de autenticação.
Porta Comandos	Insira a senha. Este campo está disponível se 'Chave pública' for selecionado como método de autenticação. Digite a porta. Insira os comandos.
Tipo de script: Telnet	O suporte a macro depende do escopo do script (consulte Escopo acima). macros de usuário personalizados são suportados.

Parâmetro		Descrição
	Nome de usuário	Digite o nome de usuário.
	Senha	Digite a senha.
	Porta	Digite a porta
	Comandos	Insira os comandos. O suporte a macro depende do escopo do script (consulte Escopo acima). macros de usuário personalizados são suportados.
	Tipo de script: IPMI Command	Insira o comando IPMI. O suporte a macro depende do escopo do script (consulte Escopo acima). macros de usuário personalizados são suportados.
Descrição		Digite uma descrição para o script.
Grupo de hosts		Selecione o grupo de hosts para o qual o script estará disponível (ou All para todos os grupos de hosts).
Grupo de usuários		Selecione o grupo de usuários para o qual o script estará disponível (ou Todos para todos os grupos de usuários). Este campo é exibido apenas se 'Ação manual do host' ou 'Ação de evento manual' está selecionada como Escopo.
Permissões de host necessárias		Selecione o nível de permissão para o grupo de hosts - Read ou Write. Somente usuários com o nível de permissão necessário terão acesso para executar o script. Este campo é exibido apenas se 'Ação manual do host' ou 'Ação manual do evento' estiver selecionado como Escopo.
Ativar confirmação		Marque a caixa de seleção para exibir uma mensagem de confirmação antes de executar o script. Esse recurso pode ser especialmente útil com operações potencialmente perigosas (como um script de reinicialização) ou que podem levar muito tempo. Esta opção é exibida apenas se 'Ação manual do host' ou 'Ação manual do evento' estiver selecionado como Escopo .
Texto de confirmação		Insira um texto de confirmação personalizado para o pop-up de confirmação ativado com a caixa de seleção acima (por exemplo, O sistema remoto será reinicializado. Tem certeza?). Para ver a aparência do texto, clique em Confirmação de teste ao lado do campo. As macros {HOST.*} e {USER.*} são compatíveis. macros de usuário personalizados são suportados. Observação: as macros não serão expandidas ao testar a mensagem de confirmação. Este campo é exibido apenas se 'Ação manual do host' ou 'Ação de evento manual' está selecionada como Escopo.

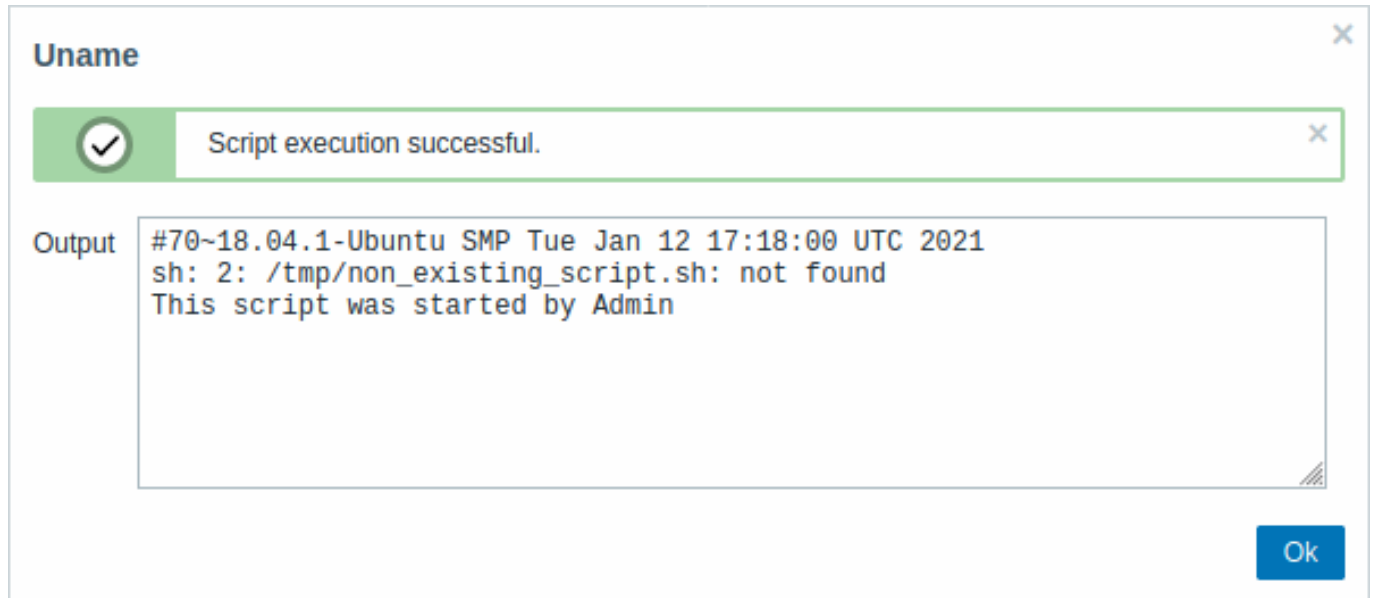
Execução e resultado do script

Os scripts executados pelo servidor Zabbix são executados pela ordem descrita em Seção **Execução do comando** incluindo a verificação do código de saída. O resultado do script será exibido em um janela pop-up que aparecerá após a execução do script.

Nota: O valor de retorno do script é a saída padrão junto com erro padrão.

Veja um exemplo de script e a janela de resultados abaixo:

```
uname -v
/tmp/non_existing_script.sh
echo "Este script foi iniciado por {USER.USERNAME}"
```



O resultado do script não exibe o script em si.

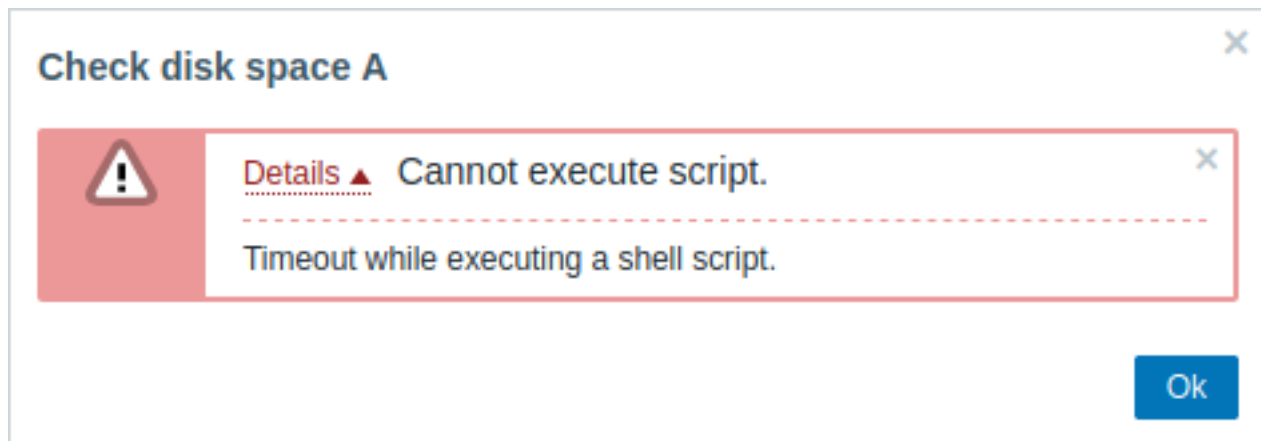
Tempo limite do script

Agente Zabbix

Você pode encontrar uma situação em que ocorre um tempo limite durante a execução de um roteiro.

Veja um exemplo de script rodando no agente Zabbix e a janela de resultados abaixo:

```
dormir 5
df -h
```



A mensagem de erro, neste caso, é a seguinte:

Tempo limite durante a execução de um script de shell.

Para evitar tal situação, é aconselhável otimizar o script em si (em vez de ajustar o parâmetro Timeout para um valor correspondente (no nosso caso, > '5') modificando o [agente Zabbix configuração](#) e [servidor Zabbix configuração](#)).

Caso ainda seja alterado o parâmetro Timeout no [agente Zabbix configuration](#) após o erro mensagem aparece:

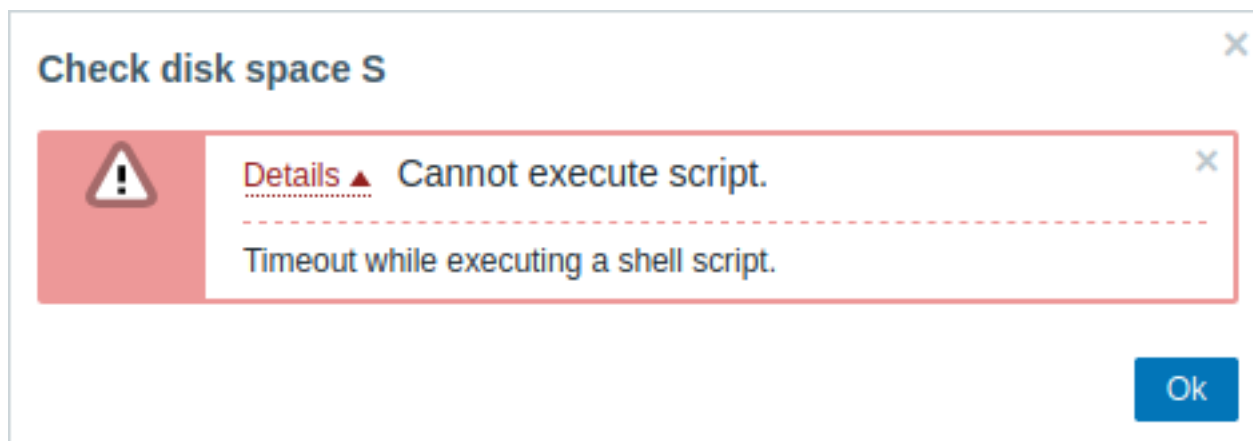
Falha ao obter valor do agente: ZBX_TCP_READ() expirou.

Significa que a modificação foi feita no [agente Zabbix configuração](#) e é obrigatório para modificar a configuração de tempo limite também no [servidor Zabbix configuração](#).

Servidor/proxy Zabbix

Veja um exemplo de script rodando no servidor Zabbix e o resultado janela abaixo:

```
dormir 11  
df -h
```



Também é aconselhável otimizar o próprio script (em vez de ajustar TrapperTimeout para um valor correspondente (no nosso caso, > '11') modificando o [servidor Zabbix configuração](#)).

9 Fila

Visão geral

Na seção Administration → Queue os itens que estão esperando para serem atualizados são exibidos.

Idealmente, quando você abre esta seção, tudo deve estar “verde”, o que significa que não itens na fila. Se todos os itens forem atualizados sem demora, haverá nenhum esperando. No entanto, devido à falta de desempenho do servidor, a conexão problemas ou problemas com os agentes, alguns itens podem atrasar e o informações são exibidas nesta seção. Para mais detalhes, consulte o seção [Fila](#).

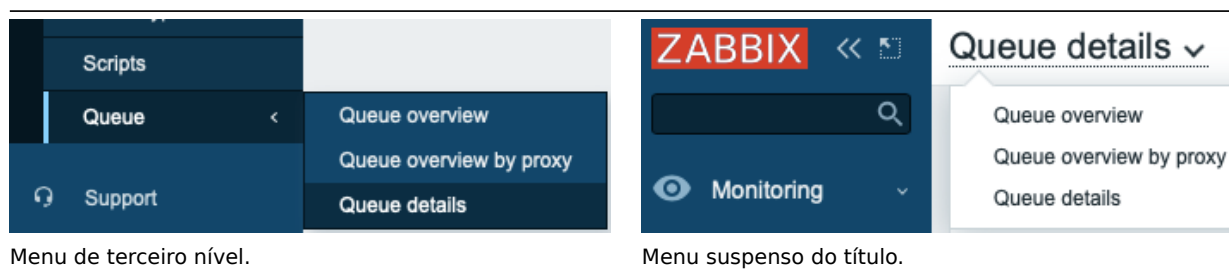
Note:

A fila está disponível apenas se o servidor Zabbix estiver correndo.

A seção Administração → Fila contém as seguintes páginas:

- Visão geral da fila — exibe a fila por tipo de item;
- Visão geral da fila por proxy — exibe a fila por proxy;
- Detalhes da fila — exibe uma lista de itens atrasados.

A lista de páginas disponíveis aparece ao pressionar Queue no Seção do menu Administração. Também é possível alternar entre páginas usando um menu suspenso de título no canto superior esquerdo.



Visão geral por tipo de item

Nesta tela é fácil localizar se o problema está relacionado a um ou vários tipos de itens.

Queue overview

Items	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Zabbix agent	1	11	1	0	0	0
Zabbix agent (active)	0	0	0	0	0	0
Simple check	0	0	0	0	0	0
SNMPv1 agent	0	0	0	0	0	0
SNMPv2 agent	0	0	0	0	0	0
SNMPv3 agent	0	0	0	0	0	0
Zabbix internal	0	0	0	0	0	0
Zabbix aggregate	0	0	0	0	0	0
External check	0	0	0	0	0	0
Database monitor	0	0	0	0	0	0
HTTP agent	0	0	0	0	0	0

Cada linha contém um tipo de item. Cada coluna mostra o número de espera itens - aguardando 5-10 segundos/10-30 segundos/30-60 segundos/1-5 minutos/5-10 minutos ou mais de 10 minutos, respectivamente.

Visão geral por proxy

Nesta tela é fácil localizar se o problema está relacionado a um dos os proxies ou o servidor.

Queue overview by proxy

Proxy	5 seconds	10 seconds	30 seconds	1 minute	5 minutes	More than 10 minutes
Remote proxy	0	8	11	0	0	0
Server	0	0	0	0	0	0

Total: 2

Cada linha contém um proxy, com o servidor por último na lista. Cada coluna mostra o número de itens em espera - aguardando 5-10 segundos/10-30 segundos/30-60 segundos/1-5 minutos/5-10 minutos ou mais de 10 minutos respectivamente.

Lista de itens em espera

Nesta tela, cada item em espera é listado.

Queue details

Scheduled check	Delayed by	Host	Name	Proxy
2019-09-02 11:46:40	58s	My host	CPU idle time	Remote proxy
2019-09-02 11:46:41	57s	My host	CPU interrupt time	Remote proxy
2019-09-02 11:46:42	56s	My host	CPU iowait time	Remote proxy
2019-09-02 11:46:43	55s	My host	CPU nice time	Remote proxy
2019-09-02 11:46:44	54s	My host	CPU softirq time	Remote proxy
2019-09-02 11:46:45	53s	My host	CPU steal time	Remote proxy
2019-09-02 11:46:46	52s	My host	CPU system time	Remote proxy

Dados exibidos:

Coluna	Descrição
Cheque agendado	É exibida a hora de vencimento do cheque.
Atrasado por	A duração do atraso é exibida.
Host	Host do item é exibido.
Nome	O nome do item em espera é exibido.
Proxy	O nome do proxy é exibido, se o host for monitorado por proxy.

Possíveis mensagens de erro

Você pode encontrar uma situação em que nenhum dado é exibido e o aparece a seguinte mensagem de erro:

Details	Cannot display item queue.
Permission denied.	

A mensagem de erro neste caso é a seguinte:

Não é possível exibir a fila de itens. Permissão negada


Isso acontece quando os parâmetros de configuração do PHP \$ZBX_SERVER_PORT ou \$ZBX_SERVER em zabbix.conf.php aponta para o servidor Zabbix existente que usa banco de dados diferente.

3 Configurações do usuário

Visão geral

Dependendo das permissões da função do usuário, a seção Configurações do usuário pode conter as seguintes páginas:

- Perfil de usuário - para customizar alguns recursos do frontend do Zabbix;
- Tokens de API - para gerenciar tokens de API atribuídos ao usuário atual.

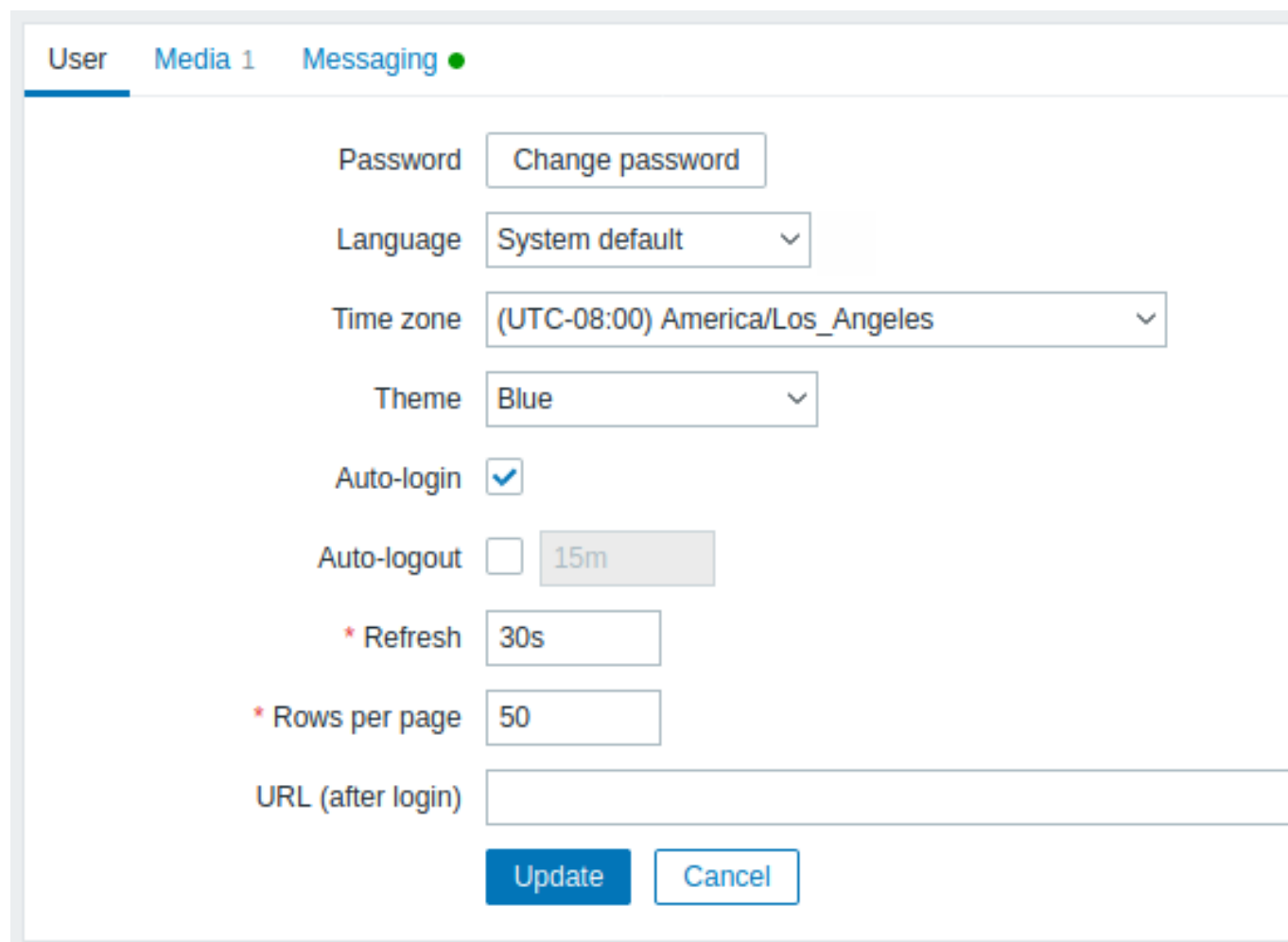
A lista de páginas disponíveis aparece ao pressionar o botão  ícone do usuário próximo à parte inferior do menu Zabbix (não disponível para um usuário convidado). Isto também é possível alternar entre as páginas usando um menu suspenso de título no o canto superior esquerdo.



1 Perfil de usuário

A seção **Perfil do usuário** oferece opções para definir a interface personalizada idioma, tema de cores, número de linhas exibidas nas listas, etc. as alterações feitas aqui serão aplicadas apenas ao usuário atual.

A guia **Usuário** permite definir várias preferências do usuário.



Parâmetro	Descrição
Senha	Clique no link para exibir dois campos para inserir uma nova senha.
Idioma	<p>Selecione o idioma da interface de sua escolha ou selecione Padrão do sistema para usar as configurações padrão do sistema.</p> <p>Para obter mais informações, consulte Instalação de idiomas front-end adicionais.</p>
Fuso horário	<p>Selecione o fuso horário para substituir o [fuso horário] global (/manual/web_interface/time_zone#overview) no nível do usuário ou selecione Padrão do sistema para usar as configurações globais de fuso horário.</p>
Tema	<p>Selecione um tema de cor especificamente para o seu perfil:</p> <p>Padrão do sistema - use as configurações padrão do sistema</p> <p>Azul - tema azul padrão</p> <p>Escuro - alternativa tema escuro</p> <p>Luz de alto contraste - tema claro com alto contraste</p> <p>Escuro de alto contraste - tema escuro com alto contraste</p>
Auto-login	<p>Marque esta caixa de seleção para que o Zabbix se lembre de você e faça seu login automaticamente por 30 dias. Os cookies do navegador são usados para isso.</p>
Auto-logout	<p>Com esta caixa de seleção marcada, você será desconectado automaticamente, após a quantidade definida de segundos (mínimo de 90 segundos, máximo de 1 dia).</p> <p>Sufixos de tempo são suportado, por exemplo 90s, 5m, 2h, 1d. Observe que esta opção não funcionará:</p> <p>* Quando as páginas do menu Monitoramento executam atualizações de informações de segundo plano. Caso as páginas que atualizam dados em um intervalo de tempo específico (painéis, gráficos, dados mais recentes, etc.) sejam deixadas abertas, a duração da sessão é estendida, respectivamente, desativando o recurso de logout automático;</p> <p>* Se fizer login com o botão Lembrar-me para Opção de 30 dias marcada.</p> <p>O logout automático pode aceitar 0, o que significa que o logout automático é desativado após a atualização das configurações do perfil.</p>
Atualizar	<p>Você pode definir com que frequência as informações nas páginas serão atualizadas no menu Monitoramento, exceto o Painel, que usa seus próprios parâmetros de atualização para cada widget.</p> <p>Sufixos de tempo são suportados, por exemplo 30s, 5m, 2h, 1d.</p>
Linhas por página	<p>Você pode definir quantas linhas serão exibidas por página nas listas. Menos linhas (e menos registros a serem exibidos) significam tempos de carregamento mais rápidos.</p>
URL (após o login)	<p>Você pode definir um URL específico para ser exibido após o login. Em vez do padrão Monitoramento → Painel pode ser, por exemplo, a URL de Monitoramento → Acionadores.</p>

A guia **Mídia** permite especificar a **mídia detalhes** para o usuário, como os tipos, os endereços a serem usados e quando usá-los para entregar notificações.

User

Media 1

Messaging ●

Media

Type	Send to	When active	Use if severity
Email	user@company.com	1-7,00:00-24:00	N I W A H D
Add		

Note:

Apenas **administrador level** usuários (Admin e Super admin) podem alterar seus próprios detalhes de mídia.

A guia **Mensagens** permite que você defina **global notificações**.

2 Tokens de API

A seção de tokens de API permite visualizar tokens atribuídos ao usuário, editar detalhes do token e **criar novo tokens**. Esta seção só está disponível para um usuário se a ação Gerenciar tokens de API é permitido no **usuário função** definições.

API tokens

					Create API token
					Filter
<input type="checkbox"/>	Name ▲	Expires at	Created at	Last accessed at	Status
<input type="checkbox"/>	Token 1	Never	2021-01-22 18:58:11	Never	Enabled
<input type="checkbox"/>	Token 2	2021-01-26 00:00:00	2021-01-22 16:13:03	Never	Enabled
					Displaying 2 of 2 found

Você pode filtrar tokens de API por nome, data de expiração ou status (ativado desativado). Clique no status do token na lista para rapidamente habilitar/desabilitar um token. Você também pode ativar/desativar tokens em massa selecionando-os na lista e clicando no botão Habilitar/Desabilitar botões abaixo da lista.

::: não importante Os usuários não podem ver Auth token valor dos tokens atribuídos a eles no Zabbix. Token de autenticação valor é exibido apenas uma vez - imediatamente após a criação de um token. Se isso foi perdido, o token deve ser regenerado. :::

1 Notificações globais

Visão geral

As notificações globais são uma maneira de exibir problemas que estão atualmente acontecendo na tela em que você está no frontend do Zabbix.

Sem notificações globais, trabalhando em algum outro local que não Problemas ou o Painel não mostraria nenhuma informação sobre questões que estão acontecendo atualmente. As notificações globais serão exibidas essas informações, independentemente de onde você estiver.

As notificações globais envolvem mostrar uma mensagem e **reproduzir um som**.

::: não importante A reprodução automática de sons pode estar desabilitada nos últimos versões do navegador por padrão. Neste caso, você precisa alterar isso configuração manualmente. :::

Configuração

As notificações globais podem ser habilitadas por usuário na guia Mensagens do **configuração do perfil**.

User
Media 1
Messaging ●

Frontend messaging ☒

Message timeout

Play sound

Trigger severity

☒ Recovery

☒ Not classified

☒ Information

☒ Warning

☒ Average

☒ High

☒ Disaster

Show suppressed problems ☐



Parâmetro	Descrição
Mensagens frontais	Marque a caixa de seleção para habilitar notificações globais.
Tempo limite da mensagem	Você pode definir por quanto tempo a mensagem será exibida. Por padrão, as mensagens permanecerão na tela por 60 segundos. Sufixos de tempo são suportados, por exemplo, 30s, 5m, 2h, 1d.
Tocar som	Você pode definir por quanto tempo o som será reproduzido. Uma vez - o som é reproduzido uma vez e totalmente. 10 segundos - o som é repetido por 10 segundos. Tempo limite da mensagem - o som é repetido enquanto a mensagem está visível.
Gravidade do gatilho	Você pode definir as gravidades do gatilho para as quais as notificações e sons globais serão ativados. Você também pode selecionar os sons apropriados para várias gravidades. Se nenhuma gravidade for marcada, nenhuma mensagem será exibida. Além disso, as mensagens de recuperação serão exibidas apenas para as gravidades marcadas. Portanto, se você marcar Recovery e Disaster, notificações globais serão exibidas para os problemas e as recuperações de gatilhos de gravidade de desastres.
Mostrar problemas suprimidos	Marque a caixa de seleção para exibir notificações de problemas que de outra forma seriam suprimidos (não mostrados) devido à manutenção do host.

Mensagens globais exibidas

À medida que as mensagens chegam, elas são exibidas em uma seção flutuante no lado direito. Esta seção pode ser reposicionada livremente arrastando o cabeçalho da seção.



Para esta seção, vários controles estão disponíveis:


-  O botão **Soneca** silencia o som do alarme ativo no momento;
-  O botão **Mute/Unmute** alterna entre tocar e não tocar o alarme soa em tudo.


2 Som nos navegadores

Visão geral

O som é usado em **global notificações**.

Para que os sons sejam reproduzidos no frontend do Zabbix, mensagens no frontend deve estar habilitado na aba Mensagens do perfil do usuário, com todos os acionadores gravidades verificadas, e os sons também devem ser ativados no global janela pop-up de notificação.

Se por algum motivo o áudio não puder ser reproduzido no dispositivo, o  no janela pop-up de notificação global permanecerá permanentemente no "mudo" estado e a mensagem "Não é possível suportar o áudio de notificação para este device." será exibido

ao passar o mouse sobre o .

Os sons, incluindo os clipes de áudio padrão, são suportados no formato MP3 só.

Os sons do frontend do Zabbix foram testados com sucesso nos últimos Navegadores Firefox/Opera no Linux e Chrome, Firefox, Microsoft Edge, Navegadores Opera e Safari no Windows.

Attention:

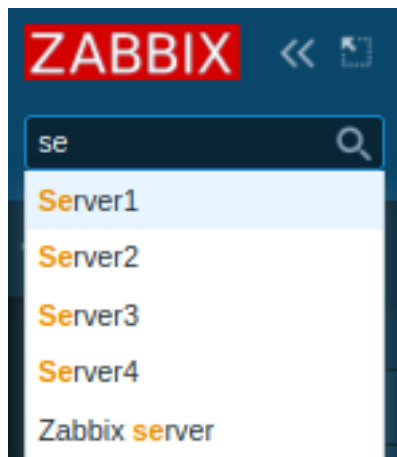
A reprodução automática de sons pode estar desabilitada nos últimos versões do navegador por padrão. Neste caso, você precisa alterar isso configuração manualmente.

4 Pesquisa global

É possível pesquisar o frontend do Zabbix por hosts, grupos de hosts e modelos.

A caixa de entrada de pesquisa está localizada abaixo do logotipo do Zabbix no menu. O a pesquisa pode ser iniciada pressionando

Enter ou clicando no botão  pesquisa ícone.



Se houver um host que contenha a string inserida em qualquer parte do nome, uma lista suspensa aparecerá, listando todos esses hosts (com a correspondência parte destacada em laranja). A lista suspensa também listará um host se esse o nome visível do host corresponde ao nome técnico inserido como pesquisa corda; o host correspondente será listado, mas sem nenhum destaque.

Atributos pesquisáveis

Os hosts podem ser pesquisados pelas seguintes propriedades:

- Nome de anfitrião
- Nome visível
- Endereço de IP
- Nome DNS

Os grupos de hosts podem ser pesquisados por nome. Especificando um grupo de hosts pai seleciona implicitamente todos os grupos de hosts aninhados.

Os modelos podem ser pesquisados por nome ou nome visível. Se você pesquisar por um nome que é diferente do nome visível (de um template/host), em os resultados da pesquisa é exibido abaixo do nome visível em parênteses.

Procurar Resultados

Os resultados da pesquisa consistem em três blocos separados para hosts, grupos de hosts e modelos.

≡ Search: Zabbix server

Hosts

Host

IP

DNS

Monitoring

Configuration

Zabbix server

127.0.0.1

Latest data

Problems

Graphs

Dashboards

Web

Items 141

Triggers 64

Graphs 27

Discovery 3

Web 1

Displaying 1 of 1 found

Host groups

Host group

Monitoring

Configuration

Zabbix servers

Latest data

Problems

Web

Hosts 1

Templates

Displaying 1 of 1 found

Templates

Template

Configuration

Template App Remote Zabbix server

Items 47

Triggers 34

Graphs 6

Dashboards 1

Discovery

Web

Template App Zabbix Server

Items 46

Triggers 34

Graphs 6

Dashboards 1

Discovery

Web

Displaying 2 of 2 found

É possível recolher/expandir cada bloco individual. A contagem de entrada é exibido na parte inferior de cada bloco, por exemplo, Exibindo 13 de 13 encontrados. O total de entradas exibidas em um bloco é limitado a 100.

Cada entrada fornece links para dados de monitoramento e configuração. Veja o [lista completa](#) de ligações.

Para todos os dados de configuração (como itens, acionadores, gráficos) a quantidade de entidades encontradas é exibido por um número ao lado do nome da entidade, em cinza. **Observe** que, se houver zero entidades, nenhum número será exibido.

Hosts habilitados são exibidos em azul, hosts desabilitados em vermelho.

Links disponíveis

Para cada entrada estão disponíveis os seguintes links:

- Anfitriões
 - Monitoramento

- * Dados mais recentes
- * Problemas
- * Gráficos
- * Painéis de hospedagem
- * Cenários da Web
- Configuração
 - * Itens
 - * Gatilhos
 - * Gráficos
 - * Regras de descoberta
 - * Cenários da Web
- Grupos de anfitriões
 - Monitoramento
 - * Dados mais recentes
 - * Problemas
 - * Cenários da Web
 - Configuração
 - * Anfitriões
 - * Modelos
- Modelos
 - Configuração
 - * Itens
 - * Gatilhos
 - * Gráficos
 - * Painéis de modelos
 - * Regras de descoberta
 - * Cenários da Web

5 Modo de manutenção de front-end

Visão geral

A interface web do Zabbix pode ser temporariamente desativada para bloquear o acesso a ela. Isso pode ser muito útil para evitar que o banco de dados de configuração do Zabbix seja alterado por seus usuários, protegendo a integridade do banco.

O banco de dados do Zabbix pode ser parado durante as tarefas de manutenção enquanto a interface web está em modo de manutenção.

Apenas usuários de determinados IPs pré-definidos conseguirão utilizar a interface web durante o modo de manutenção.

Configuração

Para habilitar o modo de manutenção será necessário alterar o arquivo `maintenance.inc.php` que fica localizado no diretório `/conf` dentro da interface web do Zabbix, isso pode ser feito ao alterar as seguintes linhas:

```
// Modo de manutenção.
define('ZBX_DENY_GUI_ACCESS', 1);

// Array de endereços IP que tem permissão de conectar-se à interface web (opcional).
$ZBX_GUI_ACCESS_IP_RANGE = array('127.0.0.1');

// Mensagem de aviso de interface bloqueada para manutenção (opcional).
$ZBX_GUI_ACCESS_MESSAGE = 'We are upgrading MySQL database till 15:00. Stay tuned...';
```

Parâmetro	Detalhes
ZBX_DENY_GUI_ACCESS	Controla o modo de manutenção: 1 - modo de manutenção ativo, inativo com qualquer outro valor
ZBX_GUI_ACCESS_IP_RANGE	Array com endereços IP que possuem permissão de conectar-se à interface web (opcional). Exemplo: <code>array('192.168.1.1', '192.168.1.2')</code>

Parâmetro	Detalhes
ZBX_GUI_ACCESS_MESSAGE	Mensagem de aviso de interface bloqueada para manutenção (opcional).

Note:

Mostly the `maintenance.inc.php` file is located in `/conf` of Zabbix HTML document directory on the web server. However, the location of the directory may differ depending on the operating system and a web server it uses.

For example, the location for:

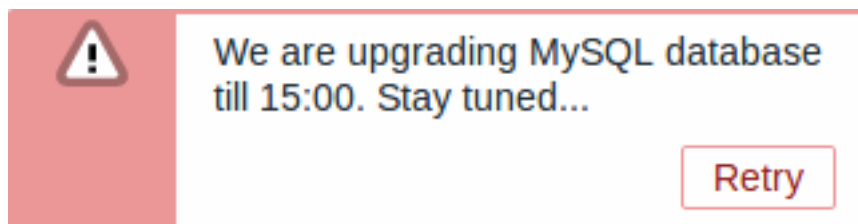
- SUSE and RedHat is `/etc/zabbix/web/maintenance.inc.php`.
- Debian-based systems is `/usr/share/zabbix/conf/`.

See also [Copying PHP files](#).

Parameter	Details
ZBX_DENY_GUI_ACCESS	Enable maintenance mode: 1 - maintenance mode is enabled, disabled otherwise
ZBX_GUI_ACCESS_IP_RANGE	Array of IP addresses, which are allowed to connect to frontend (optional). For example: <code>array('192.168.1.1', '192.168.1.2')</code>
ZBX_GUI_ACCESS_MESSAGE	Message you can enter to inform users about the maintenance (optional).

Apresentação

A tela a seguir será apresentada quando usuários, que não tenham seu IP definido no parâmetro **ZBX_GUI_ACCESS_IP_RANGE**, tentarem acessar a interface web durante o modo de manutenção. A tela será atualizada a cada 30 segundos para que o funcionamento da interface volte ao normal sem precisar de intervenção direta dos usuários.



6 Parâmetros da página

Visão geral

A maioria das páginas da interface web do Zabbix suporta vários parâmetros HTTP GET que controlar o que será exibido. Eles podem ser passados especificando parâmetro=valor após o URL, separado do URL por um ponto de interrogação (?) e um do outro por e comercial (&).

Monitoramento → Problemas

Os seguintes parâmetros são suportados:

- `show` - opção de filtro "Show": 1 - problemas recentes, 2 - todos, 3 - em estado do problema
- `name` - opção de filtro "Problema": string de forma livre
- `severities` - opção de filtro "Severity": array de selecionado gravidades em um formato `'severities[*]=*'` (substitua * por nível de gravidade): 0 - não classificado, 1 - informação, 2 - aviso, 3 - médio, 4 - alto, 5 - desastre
- `inventário` - opção de filtro "Inventário do host": array de inventário campos: [campo], [valor]
- `evaltype` - opção de filtro "Tags", estratégia de filtragem de tags: 0 - E/ou, 2 - ou
- `tags` - opção de filtro "Tags": array de tags definidas: [tag], [operador], [valor]
- `show_tags` - opção de filtro "Mostrar tags": 0 - nenhum, 1 - um, 2 - dois, 3 - três
- `tag_name_format` - opção de filtro "Nome da tag": 0 - nome completo, 1 - encurtado, 2 - nenhum
- `tag_priority` - opção de filtro "Prioridade de exibição de tags": string separada por vírgula de prioridade de exibição de tag
- `show_suppressed` - opção de filtro "Mostrar problemas suprimidos": deve seja `'show_suppressed=1'` para mostrar
- `unacknowledged` - opção de filtro "Mostrar apenas não confirmado": deve ser `'não reconhecido=1'` para mostrar
- `compact_view` - opção de filtro "Visualização compacta": deve ser `'compact_view=1'` para mostrar

- `highlight_row` - opção de filtro "Realçar linha inteira" (use problema cor como cor de fundo para cada linha de problema): deve ser '1' para realçar; pode ser definido apenas quando 'compact_view' está definido
- `filter_name` - opção de propriedades do filtro "Nome": string de forma livre
- `filter_show_counter` - opção de propriedades do filtro "Mostrar número de registros": 1 - mostrar, 0 - não mostrar
- `filter_custom_time` - opção de propriedades do filtro "Definir hora personalizada period": 1 - definido, 0 - não definido
- `sort` - coluna de classificação: relógio, host, gravidade, nome
- `sortorder` - ordem de classificação ou resultados: DESC - decrescente, ASC - ascendente
- `age_state` - opção de filtro "Idade menor que": deve ser 'age_state=1' para habilitar 'age'. É usado apenas quando 'show' é igual a 3.
- `idade` - opção de filtro "Idade menor que": dias
- `groupids` - opção de filtro "Grupos de hosts": array de IDs de grupos de hosts
- `hostids` - opção de filtro "Hosts": array de IDs de host
- `triggerids` - opção de filtro "Triggers": array de IDs de trigger
- `show_timeline` - opção de filtro "Mostrar linha do tempo": deve ser 'show_timeline=1' para mostrar
- `details` - opção de filtro "Mostrar detalhes": deve ser 'details=1' para exposição
- `from` - início do intervalo de datas, pode ser 'relativo' (por exemplo: now-1m). É usado somente quando 'filter_custom_time' for igual a 1.
- `to` - fim do intervalo de datas, pode ser 'relativo' (por exemplo: agora-1m). É usado somente quando 'filter_custom_time' for igual a 1.

Modo Kiosk

O modo kiosk em páginas frontend suportadas pode ser ativado usando URL parâmetros. Por exemplo, em painéis:

- `/zabbix.php?action=dashboard.view&kiosk=1` - ativa o modo kiosk
- `/zabbix.php?action=dashboard.view&kiosk=0` - ativa o modo normal

Apresentação de slides

É possível ativar uma apresentação de slides no painel:

- `/zabbix.php?action=dashboard.view&slideshow=1` - ativar apresentação de slides

7 Definições

Visão geral

Embora muitas coisas no frontend possam ser configuradas usando o frontend em si, algumas personalizações atualmente só são possíveis editando um arquivo de definições.

Este arquivo é `defines.inc.php` localizado em `/include` do HTML do Zabbix diretório de documentos.

Parâmetros

Parâmetros neste arquivo que podem ser de interesse dos usuários:

- `ZBX_MIN_PERIOD`

Período mínimo do gráfico, em segundos. Um minuto por padrão.

- `GRAPH_YAXIS_SIDE_DEFAULT`

Localização padrão do eixo Y em gráficos simples e valor padrão para soltar caixa para baixo ao adicionar itens a gráficos personalizados. Valores possíveis: 0 - esquerda, 1 - certo.

Padrão: 0

- `ZBX_SESSION_NAME` (disponível desde 4.0.0)

String usada como o nome do cookie de sessão de frontend do Zabbix.

Padrão: `zbx_sessionid`

- `ZBX_DATA_CACHE_TTL` (disponível desde 5.2.0)

Tempo limite TTL em segundos usado para invalidar o cache de dados de **Vault resposta**. Defina 0 para desabilitar a resposta do Vault cache.

Padrão: 60

8 Criando seu próprio tema

Visão geral

Por padrão, o Zabbix fornece vários temas predefinidos. Você pode seguir o procedimento passo a passo fornecido aqui para criar seu tema. Sinta-se à vontade para compartilhar o resultado do seu trabalho com a comunidade Zabbix se você criou algo legal.

Passo 1

Para definir seu próprio tema, você precisará criar um arquivo CSS e salvá-lo em a pasta `assets/styles/` (por exemplo, `custom-theme.css`). Você pode ou copie os arquivos de um tema diferente e crie seu tema baseado nele ou comece do zero.

Passo 2

Adicione seu tema à lista de temas retornados pelo `APP::getThemes()` método. Você pode fazer isso substituindo o método `ZBase::getThemes()` em a aula de APP. Isso pode ser feito adicionando o seguinte código antes do chave de fechamento em `include/classes/core/APP.php`:

```
função estática pública getThemes() {  
    return array_merge(parent::getThemes(), [  
        'tema personalizado' => _('tema personalizado')  
    ]);  
}
```

::: não importante Observe que o nome especificado no primeiro par de aspas deve corresponder ao nome do arquivo do tema sem extensão. :::

Para adicionar vários temas, basta listá-los sob o primeiro tema, por exemplo:

```
função estática pública getThemes() {  
    return array_merge(parent::getThemes(), [  
        'tema personalizado' => _('tema personalizado'),  
        'outro tema' => _('Outro tema'),  
        'onemoretheme' => _('Mais um tema')  
    ]);  
}
```

Observe que todo tema, exceto o último, deve ter uma vírgula à direita.

Para alterar as cores do gráfico, a entrada deve ser adicionada no `graph_theme` tabela de banco de dados.

Etapa 3

Ative o novo tema.

No frontend do Zabbix, você pode definir este tema para ser o padrão ou altere seu tema no perfil do usuário.

Aproveite o novo visual e toque!

9 Modo de depuração

Visão geral

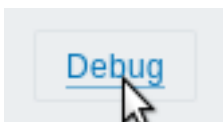
O modo de depuração pode ser usado para diagnosticar problemas de desempenho com páginas frontend.

Configuração

O modo de depuração pode ser ativado para usuários individuais que pertencem a um grupo de usuário.

- ao configurar um **usuário group**;
- ao visualizar configurado **usuário groups**.

Quando o Modo de depuração está habilitado para um grupo de usuários, seus usuários verão um botão Depurar no canto inferior direito da janela do navegador:



Clicar no botão Debug abre uma nova janela abaixo da página conteúdo que contém as estatísticas SQL da página, juntamente com um lista de chamadas de API e instruções SQL individuais:

```
***** Script profiler *****
Total time: 0.249825
Total SQL time: 0.139814
SQL count: 143 (selects: 117 | executes: 26)
Peak memory usage: 6M
Memory limit: 128M

1. hostgroup.get [latest.php:124]

Parameters:          Result:
Array                Array
(                    (
    [output] => Array    [4] => Array
        (                (
            [0] => groupid    [groupid] => 4
        )                )
    )                    )

```

Hide debug

Em caso de problemas de desempenho com a página, esta janela pode ser usada para procurar a causa raiz do problema. Habilitado Modo de depuração afeta negativamente o frontend atuação.

10 Cookies usados pelo Zabbix

Visão geral

Esta página fornece uma lista de cookies usados pelo Zabbix.

[Nome|Descrição|Valores|Expira/Max-Age|HttpOnly⁵|Protegido⁶]|----|-----|-----|-----|-----| |ZBX_SESSION_NAME|Dados da sessão de frontend Zabbix, armazenados como JSON codificados por base64|<|Session (expira quando a sessão de navegação termina)|+|+ (somente se HTTPS estiver habilitado em um servidor web)| |tab|Número da guia ativa; este cookie é usado apenas em páginas com várias guias (por exemplo, página de configuração Host, Trigger ou Action) e é criado quando um usuário navega de uma guia primária para outra guia (como Tags ou Dependencias tab).

0 é usado para a aba principal.|Exemplo: 1|Sessão (expira quando a sessão de navegação termina)|-|-| |browserwarning_ignore|Se um aviso sobre o uso de um navegador desatualizado deve ser ignorado.|sim|Sessão (expira quando a sessão de navegação termina)|-|-| |system-message-ok|Uma mensagem para mostrar assim que a página for recarregada.|Mensagem de texto simples|Sessão (expira quando a sessão de navegação terminar) ou assim que a página for recarregada.|+|-| |system-message-error|Uma mensagem de erro para mostrar assim que a página for recarregada.|Mensagem de texto simples|Sessão (expira quando a sessão de navegação terminar) ou assim que a página for recarregada|+|-|

Note:

Forçando o sinalizador 'HttpOnly' nos cookies do Zabbix por um servidor web diretiva não é suportada.

11 Fuso horário

Visão Geral

O fuso horário do frontend pode ser definido globalmente no frontend e ajustado para usuários individuais.

⁵De acordo com [especificação](#) estas são voltagens nos pinos do chip e, de um modo geral, podem precisar dimensionamento.

⁶**Secure** indica que o cookie só deve ser transmitido por uma conexão HTTPS segura do cliente. Quando definido como 'true', o cookie só será definido se existir uma conexão segura.

Se o Sistema for selecionado, o fuso horário do servidor web será usado para o frontend (incluindo o valor de 'date.timezone' do php.ini, se configurado), enquanto o servidor Zabbix usará o fuso horário da máquina em que está sendo executado.

Note:

O servidor Zabbix usará apenas o fuso horário global/usuário especificado ao expandir macros em notificações (por exemplo, o {EVENT.TIME} pode expandir para um fuso horário diferente por usuário) e para o limite de tempo quando as notificações são enviadas (consulte a configuração "Quando ativo" [media configuration](#)).

Configuração

Fuso horário mundial:

- pode ser definido manualmente ao [Instalar](#) o frontend
- pode ser alterado em Administração → Geral → [GUI](#)

Fuso horário em nível de usuário:

- pode ser definido quando [Configurando/Atualizando](#) um usuário
- pode ser definido por cada usuário em seu [perfil de usuário](#)

13 Redefinindo senha

Visão geral Esta seção descreve os passos para redefinir senhas de usuários no Zabbix.

Passos Consulte o administrador do Zabbix se você esqueceu sua senha do Zabbix e não consegue fazer login.

Um usuário superadministrador pode alterar as senhas de todos os usuários no [formulário de configuração](#).

Se um superadministrador esqueceu sua senha e não consegue fazer login, a seguinte consulta SQL deve ser executada para aplicar a senha padrão ao usuário superadministrador (substitua 'Admin' pelo nome de usuário do superadministrador apropriado):

```
ATUALIZAR usuarios SET passwd = '$2a$10$ZXIvHAEP2ZM.dLXTm6uPHOMV1ARXX7cqjbhM6Fn0cANzkCQBwPmR5' WHERE nome
```

Após executar esta consulta, a senha do usuário será definida como zabbix. Certifique-se de alterar a senha padrão no primeiro login.

19 API

Visão Geral A API do Zabbix permite que você recupere e modifique programaticamente a configuração do Zabbix e fornece acesso a dados históricos. Isto é amplamente utilizado para:

- Criar novos aplicativos para trabalhar com Zabbix;
- Integrar o Zabbix com softwares de terceiros;
- Automatizar tarefas de rotina.

A API do Zabbix é uma API baseada na web e é enviada como parte do frontend web. Ela usa o protocolo JSON-RPC 2.0, o que significam 2 coisas:

- A API consiste em um conjunto de métodos separados;
- Solicitações e respostas entre os clientes e a API são codificadas usando o formato JSON.

Mais informações sobre o protocolo e JSON podem ser encontradas no [JSON-RPC 2.0 specification](http://www.jsonrpc.org/specification) e [JSON format homepage](#).

Para mais informações sobre como integrar a funcionalidade de Zabbix em seus aplicativos Python, consulte a biblioteca [zabbix_utils](#) para API do Zabbix.

Estrutura A API consiste em vários métodos que são nominalmente agrupados em APIs separadas. Cada um dos métodos desempenha uma tarefa específica. Por exemplo, o método `host.create` pertence ao API `host` e é usado para criar novos hosts. Historicamente, as APIs as vezes são chamadas de "classes".

Note:

A maioria das APIs contém ao menos quatro métodos: `get`, `create`, `update` e `delete` para recuperar, criar, atualizar e excluir dados, respectivamente. Contudo, algumas APIs podem oferecer um conjunto de métodos totalmente diferente.

Desenvolvendo requisições Uma vez que você tenha configurado o frontend, você pode usar requisições de HTTP remoto para requisitar a execução da API. Para isso, você precisará enviar requisições HTTP POST para o arquivo `api_jsonrpc.php` localizado no diretório frontend. Por exemplo, se sua instalação do frontend Zabbix está em `http://example.com/zabbix`, a requisição HTTP para o método pode ser algo como a seguir:

```
POST http://example.com/zabbix/api_jsonrpc.php HTTP/1.1
Content-Type: application/json-rpc
```

```
{
  "jsonrpc": "2.0",
  "method": "apiinfo.version",
  "id": 1,
  "auth": null,
  "params": {}
}
```

A requisição deve ter o header `Content-Type` configurado para um desses valores: `application/json-rpc`, `application/json` ou `application/jsonrequest`.

Exemplo de workflow A seção seguinte irá guiá-lo com alguns exemplos de usos mais detalhados.

Autenticação Para acessar qualquer dado no Zabbix, você precisa:

- usar uma API existente **API token** (criada no frontend Zabbix ou usar a API **Token API**);
- usar um token de autenticação obtido com o método `user.login`.

Por exemplo, se você quisesse obter um novo token de autenticação ao fazer login como usuário Admin padrão, a solicitação JSON seria assim:

```
{
  "jsonrpc": "2.0",
  "method": "user.login",
  "params": {
    "username": "Admin",
    "password": "zabbix"
  }
}
```

```

},
"id": 1,
"auth": null
}

```

Vamos dar uma olhada mais de perto ao objeto solicitado. Ele tem as seguintes propriedades:

- `jsonrpc` - versão do protocolo JSON-RPC usada pela API; a API do Zabbix implementa a versão 2.0 de JSON-RPC;
- `method` - como método da API é chamado;
- `params` - os parâmetros serão passados para o método da API;
- `id` - um identificador arbitrário da solicitação;
- `auth` - um token de autenticação do usuário; enquanto não temos um, ele é definido como `null`.

Se você forneceu as credenciais corretamente, a resposta a ser retornada pela API irá conter um token de autenticação do usuário:

```

{
  "jsonrpc": "2.0",
  "result": "0424bd59b807674191e7d77572075f33",
  "id": 1
}

```

O objeto da resposta, por sua vez, contém as seguintes propriedades:

- `jsonrpc` - novamente, a versão do protocolo do JSON-RPC;
- `result` - o dado retornado pelo método;
- `id` - identificador da solicitação correspondente.

Recuperando hosts Agora nós temos um token de autenticação válido que pode ser utilizado para acessar para acessar os dados no Zabbix. We now have a valid user authentication token that can be used to access the data in Zabbix. Por exemplos, vamos usar o método `host.get` para recuperar as IDs, nomes e interfaces de todos os hosts configurados: `hosts`:

```

{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "output": [
      "hostid",
      "host"
    ],
    "selectInterfaces": [
      "interfaceid",
      "ip"
    ]
  },
  "id": 2,
  "auth": "0424bd59b807674191e7d77572075f33"
}

```

Attention:

Observe que a propriedade `auth` agora está configurada com o token de autenticação que obtivemos através `douser.login`.

O objeto de resposta irá conter o dado requisitado sobre os hosts:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostid": "10084",
      "host": "Zabbix server",
      "interfaces": [
        {
          "interfaceid": "1",
          "ip": "127.0.0.1"
        }
      ]
    }
  ]
}

```

```

    }
  ],
  "id": 2
}

```

Note:

Por questões de performance, é sempre recomendável listar as propriedades do objeto que você deseja recuperar para evitar recuperar tudo.

Criando um novo item Vamos criar um novo **item** no "Zabbix server" usando os dados que obtivemos através do método anterior (`host.get`). Isso pode ser feito através do método `item.create`:

```

{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "name": "Free disk space on $1",
    "key_": "vfs.fs.size[/home/joe/,free]",
    "hostid": "10084",
    "type": 0,
    "value_type": 3,
    "interfaceid": "1",
    "delay": 30
  },
  "auth": "0424bd59b807674191e7d77572075f33",
  "id": 3
}

```

Uma resposta bem sucedida irá conter o ID do novo item criado, que poderá ser utilizado para referenciar o item nas requisições a seguir:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "24759"
    ]
  },
  "id": 3
}

```

Note:

O método `item.create`, assim como outros métodos criados, também aceita arrays de objetos e cria pode criar v vários itens com apenas uma APIcall.

Criando vários triggers Se criar métodos aceita arrays, então podemos adicionar vários triggers **triggers** assim:

```

{
  "jsonrpc": "2.0",
  "method": "trigger.create",
  "params": [
    {
      "description": "Processor load is too high on {HOST.NAME}",
      "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
    },
    {
      "description": "Too many processes on {HOST.NAME}",
      "expression": "avg(/Linux server/proc.num[],5m)>300",
    }
  ],
  "auth": "0424bd59b807674191e7d77572075f33",
  "id": 4
}

```

```
}
```

Uma resposta bem sucedida irá conter os IDs dos triggers criados recentemente:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "17369",
      "17370"
    ]
  },
  "id": 4
}
```

Atualizando um item Habilite um item, isto é, configure seu status para "0":

```
{
  "jsonrpc": "2.0",
  "method": "item.update",
  "params": {
    "itemid": "10092",
    "status": 0
  },
  "auth": "0424bd59b807674191e7d77572075f33",
  "id": 5
}
```

Uma resposta bem sucedida irá conter uma ID do item atualizado:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "10092"
    ]
  },
  "id": 5
}
```

Note:

O método `item.update`, assim como outros métodos atualizados, também aceita objetos de array e consegue atualizar vários itens com apenas um acionamento da API.

Atualizando vários triggers Habilite vários triggers, isto é, configure seu status para "0":

```
{
  "jsonrpc": "2.0",
  "method": "trigger.update",
  "params": [
    {
      "triggerid": "13938",
      "status": 0
    },
    {
      "triggerid": "13939",
      "status": 0
    }
  ],
  "auth": "0424bd59b807674191e7d77572075f33",
  "id": 6
}
```

Uma resposta bem sucedida irá conter os IDs de triggers atualizados:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "13938",
      "13939"
    ]
  },
  "id": 6
}
```

Note:

Este é o método preferido de atualização. Alguns métodos API como `host.massupdate` permitem escrever códigos mais simples. Contudo, não é recomendável utilizar esses métodos, uma vez que eles serão removidos em versões futuras do Zabbix.

Manipulação de erros Até esse ponto, tudo o que tentamos funcionou bem. Mas, o que acontece se tentarmos fazer uma chamada incorreta para a API? Vamos tentar criar outro host chamando `host.create`, mas omitindo o parâmetro obrigatório `groups`.

```
{
  "jsonrpc": "2.0",
  "método": "host.create",
  "parâmetros": {
    "host": "Servidor Linux",
    "interfaces": [
      {
        "tipo 1",
        "principal": 1,
        "useip": 1,
        "ip": "192.168.3.1",
        "dns": "",
        "porta": "10050"
      }
    ]
  },
  "id": 7,
  "auth": "0424bd59b807674191e7d77572075f33"
}
```

A resposta conterá uma mensagem de erro:

```
{
  "jsonrpc": "2.0",
  "erro": {
    "código": -32602,
    "message": "Parâmetros inválidos.",
    "data": "Nenhum grupo para host \"Servidor Linux\"."
  },
  "id": 7
}
```

Se ocorreu um erro, em vez da propriedade `result`, a resposta object conterá uma propriedade `error` com os seguintes dados:

- `code` - um código de erro;
- `message` - um breve resumo do erro;
- `data` - uma mensagem de erro mais detalhada.

Erros podem ocorrer em diferentes casos, como, por exemplo, usar entrada incorreta de valores, um tempo limite de sessão ou tentar acessar objetos inexistentes. Seu aplicativo deve ser capaz de lidar com esses tipos de erros.

Versões da API Para simplificar o versionamento da API, desde o Zabbix 2.0.4, a versão da API corresponde à versão do próprio Zabbix. Você pode usar o método `apiinfo.version` para encontrar a versão da API com a qual você está trabalhando. Isso pode ser útil para ajustar seu aplicativo para usar recursos específicos da versão.

Garantimos a compatibilidade com versões anteriores dentro de uma versão principal. Ao fazer alterações incompatíveis com versões anteriores entre versões principais, geralmente deixamos os recursos antigos como obsoletos na próxima versão, e apenas removê-los no lançamento depois disso. Às vezes, podemos remover recursos entre os principais lançamentos sem fornecer nenhuma compatibilidade. É importante que você nunca confie em nenhum recurso e migre para alternativas mais recentes o mais rápido possível.

Você pode acompanhar todas as alterações feitas na API no [registro de alterações da API](#).

Leitura adicional Agora você já sabe o suficiente para começar a trabalhar com a API do Zabbix, mas não pare aqui. Para leitura adicional, sugerimos que você dê uma olhada na [lista de APIs disponíveis](#).

referência do método

Esta seção fornece uma visão geral das funções fornecidas pela API Zabbix e irá ajudá-lo a encontrar o caminho para as classes disponíveis e métodos.

Monitoramento A API do Zabbix permite acessar o histórico e outros dados coletados durante o monitoramento.

Cluster de alta disponibilidade

Recupere uma lista de nós do servidor e seus status.

[API de cluster de alta disponibilidade](#)

História

Recupere valores históricos coletados pelos processos de monitoramento do Zabbix para apresentação ou processamento posterior.

[API de histórico](#)

Trends

Recupere valores de trends calculados pelo servidor Zabbix para apresentação ou processamento adicional.

[API de tendências](#)

Eventos

Recupere eventos gerados por triggers, descoberta de rede e outros Sistemas Zabbix para gerenciamento de situação mais flexível ou de terceiros integração de ferramentas.

[API de eventos](#)

Problemas

Recupere problemas de acordo com os parâmetros fornecidos.

[API de problema](#)

Monitoramento de serviço

Crie uma representação hierárquica de dados de serviços de negócios/infraestrutura de TI monitorados.

[API de serviço](#)

Acordo de Nível de Serviço

Definir objetivos de nível de serviço (SLO), recuperar indicadores de nível de serviço (SLI) detalhadas informações sobre o desempenho do serviço.

[API SLA](#)

Tarefas

Interaja com o gerenciador de tarefas do servidor Zabbix, criando tarefas e recuperando resposta.

[API de tarefas](#)

Configuração A API Zabbix permite que você gerencie a configuração do seu sistema de monitoramento.

Hosts and host groups

Manage host groups, hosts and everything related to them, including host interfaces, host macros and maintenance periods.

[Host API](#) | [Host group API](#) | [Host interface API](#) | [User macro API](#) | [Value map API](#) | [Maintenance API](#)

Itens

Defina os itens a serem monitorados.

[API do item](#)

Gatilhos

Configure gatilhos para notificá-lo sobre problemas em seu sistema. Gerencie as dependências de gatilho.

[API Trigger](#)

Gráficos

Edite gráficos ou itens de gráfico separados para melhor apresentação dos dados coletados.

[API de gráfico](#) | [Item do gráfico API](#)

Modelos (Templates)

Gerencie modelos e vincule-os a hosts ou a outros modelos.

[modelo API](#) | [Mapa de valores API](#)

Exportar e importar

Exporte e importe dados de configuração do Zabbix para backups de configuração, migração ou atualizações de configuração em grande escala.

[configuração API](#)

Descoberta de baixo nível (Low-level discovery)

Configure regras de descoberta de baixo nível, bem como item, gatilho e protótipos de gráfico para monitorar entidades dinâmicas.

[regra LLD API](#) | [Protótipo de item API](#) | [Protótipo de gatilho API](#) | [Protótipo de gráfico API](#) | [Protótipo do host API](#)

Correlação de eventos

Crie regras de correlação de eventos personalizados.

[Correlação API](#)

Ações e alertas

Definir ações e operações para notificar os usuários sobre determinados eventos ou executar comandos remotos automaticamente. Tenha acesso a informações sobre alertas gerados e seus receptores.

[Ação API](#) | [Alerta API](#)

Serviços

Gerencie serviços para monitoramento de nível de serviço e recupere informações de SLA detalhadas sobre qualquer serviço.

[Serviço API](#)

Dashboards

Manage dashboards and make scheduled reports based on them.

[Dashboard API](#) | [Template dashboard API](#) | [Report API](#)

Mapas

Configure mapas para criar representações dinâmicas detalhadas de sua infraestrutura de TI.

[Mapa API](#)

Monitoramento da Web

Configure cenários da web para monitorar seus aplicativos e serviços da web.

[Cenário da Web API](#)

Descoberta de rede

Gerencie regras de descoberta em nível de rede para localizar e monitorar automaticamente novos hosts. Obtenha acesso total a informações sobre serviços descobertos e hosts.

[Regra de descoberta API](#) | [Verificação de descoberta API](#) | [Host descoberto API](#) | [Serviço descoberto API](#)

Administração ¶ Com a API do Zabbix você pode alterar as definições de administração do seu sistema de monitoramento.

Users

Add users that will have access to Zabbix, assign them to user groups and grant permissions. Make roles for granular management of user rights. Track configuration changes each user has done. Configure media types and multiple ways users will receive alerts.

[User API](#) | [User group API](#) | [User role API](#) | [Media type API](#) | [Audit log API](#)

General

Change certain global configuration options.

[Autoregistration API](#) | [Icon map API](#) | [Image API](#) | [User macro API](#) | [Settings API](#) | [Housekeeping API](#)

Expressões regulares

Gerenciar expressões regulares globais.

[Expressão regular API](#)

Proxies

Gerenciar os proxies utilizados na sua configuração de monitoramento distribuído.

[API de proxy](#)

Authentication

Change authentication configuration options.

[Authentication API](#)

API Tokens

Manage authorization tokens.

[Token API](#)

Scripts

Configure and execute scripts to help you with your daily tasks.

[Script API](#)

API information Retrieve the version of the Zabbix API so that your application could use version-specific features.

[API info API](#)

Alert

Esta classe foi projetada para trabalhar com alertas.

Referência de objetos:

- [Alert](#)

Métodos disponíveis:

- [alert.get](#) - recupera alertas

> Objeto de alerta

Os objetos a seguir estão diretamente relacionados à API `alert`.

Alerta

Note:

Os alertas são criados pelo servidor Zabbix e não podem ser modificados por meio da API.

O objeto de alerta contém informações sobre se determinadas ações foram executadas com sucesso. Tem o seguinte propriedades.

|Propriedade|**Tipo**|Descrição| |-----|-----| |alertid|string|ID do alerta.| |actionid|string|ID da ação que gerou o alerta.| |alerttype|inteiro|Tipo de alerta.

Valores possíveis:
0 - mensagem;
1 - comando remoto.| |clock|timestamp|Horário em que o alerta foi gerado.| |error|string|Texto de erro se houver problemas ao enviar uma mensagem ou executar um comando.| |esc_step|integer|Etapa de escalação de ação durante a qual o alerta foi gerado.| |eventid|string|ID do evento que acionou a ação.| |mediatypeid|string|ID do tipo de mídia que foi usado para enviar a mensagem.| |mensagem|texto|Texto da mensagem. Usado para alertas de mensagens.| |retentativas|inteiro|Número de vezes que o Zabbix tentou enviar a mensagem.| |sendto|string|Endereço, nome de usuário ou outro identificador do destinatário. Usado para alertas de mensagens.| |status|inteiro|Status indicando se a operação de ação foi executada com sucesso.

Possíveis valores para alertas de mensagens:
0 - mensagem não enviada.
1 - mensagem enviada.
2 - falhou após várias tentativas.
3 - o novo alerta ainda não foi processado pelo gerenciador de alertas.

Valores possíveis para alertas de comando:
0 - comando não executado.
1 - comando executado .
2 - tentei executar o comando no agente Zabbix mas não estava disponível.| |assunto|string|Assunto da mensagem. Usado para alertas de mensagens.| |userid|string|ID do usuário para o qual a mensagem foi enviada.| |p_eventid|string|ID do evento de problema, que gerou o alerta.| |acknowledgeid|string|ID de confirmação, que gerou o alerta.|

#####alerta.obter {#manual-api-reference-alert-get}

Descrição

integer/array alert.get(object parameters)

O método permite recuperar alertas de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

|Parâmetro|**Tipo**|Descrição| |-----|-----| |alertids|string/array|Retorna apenas alertas com os IDs fornecidos.| |actionids|string/array|Retorna apenas alertas gerados pelas ações fornecidas.| |eventids|string/array|Retorna apenas os alertas gerados pelos eventos fornecidos.| |groupids|string/array|Retorna apenas alertas gerados por objetos dos grupos de hosts fornecidos.| |hostids|string/array|Retorna apenas alertas gerados por objetos dos hosts fornecidos.| |mediatypeids|string/array|Retorna apenas alertas de mensagens que usaram os tipos de mídia fornecidos.| |objectids|string/array|Retorna apenas alertas gerados pelos objetos fornecidos| |userid|string/array|Retorna apenas alertas de mensagens que foram enviadas para os usuários indicados.| |eventobject|integer|Retorna apenas alertas gerados por eventos relacionados a objetos de um determinado tipo.

Veja event "[object](#)" para obter uma lista de tipos de objetos suportados.

Padrão: 0 - trigger.| |eventsource|integer|Retorna apenas alertas gerados por eventos de um determinado tipo.

Consulte event "[source](#)" para obter uma lista de tipos de eventos suportados .

Padrão: 0 - acionar eventos.| |time_from|timestamp|Retorna apenas os alertas que foram gerados após o tempo determinado.| |time_till|timestamp|Retorna apenas os alertas que foram gerados antes do tempo determinado.| |selectHosts|query|Retorne uma propriedade [hosts](#) com dados de hosts que acionaram a operação de ação.| |selectMediatypes|query|Retorne uma propriedade [mediatypes](#) com uma matriz dos tipos de mídia que foram usados para o alerta de mensagem.| |selectUsers|query|Retorne uma propriedade [users](#) com um array dos usuários aos quais a mensagem foi endereçada.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: alertid, clock, eventid, mediatypeid, sendto e status.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos no [comentário de referência](#).| |editável|boolean|^| |excluiPesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar alertas por ID de ação

Recupera todos os alertas gerados pela ação "3".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "alert.get",
  "params": {
    "output": "extend",
    "actionids": "3"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "alertid": "1",
      "actionid": "3",
      "eventid": "21243",
      "userid": "1",
      "clock": "1362128008",
      "mediatypeid": "1",
      "sendto": "support@company.com",
      "subject": "PROBLEM: Zabbix agent on Linux server is unreachable for 5 minutes: ",
      "message": "Trigger: Zabbix agent on Linux server is unreachable for 5 minutes: \nTrigger stat",
      "status": "0",
      "retries": "3",
      "error": "",
      "esc_step": "1",
      "alerttype": "0",
      "p_eventid": "0",
      "acknowledgeid": "0"
    }
  ],
  "id": 1
}
```

Veja também

- [Host](#)
- [Tipo de mídia](#)
- [Usuário](#)

Fonte

CAAlert::get() em ui/include/classes/api/services/CAAlert.php.

Autenticação

Essa classe foi projetada para trabalhar com parâmetros de autenticação.

Objetos de referência:

- [Authentication](#)

Métodos disponíveis:

- [authentication.get](#) - recupera autenticação
- [authentication.update](#) - atualiza autenticação

> Objeto de autenticação

Os objetos a seguir estão diretamente relacionados à API authentication.

Autenticação

O objeto de autenticação tem as seguintes propriedades.

Propriedade	Tipo	Descrição
autenticação_tipo inteiro	Autenticação padrão.	 Valores possíveis: 0 - (padrão) Interno; 1 - LDAP.
http_auth_enabled inteiro	Ativar autenticação HTTP.	 Valores possíveis: 0 - (padrão) Desativar; 1 - Ativar.
http_login_form integer	Formulário de login padrão.	 Valores possíveis: 0 - (padrão) Formulário de login do Zabbix; 1 - Formulário de login HTTP.
http_strip_domains string	Remover nome de domínio.	
http_case_sensitive inteiro	Login sensível a maiúsculas e minúsculas HTTP.	 Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.
ldap_configured inteiro	Ativar autenticação LDAP.	 Valores possíveis: 0 - Desativar; 1 - (padrão) Ativar.
ldap_host string	host LDAP.	
ldap_port inteiro	porta LDAP.	
ldap_base_dn string	DN base LDAP.	
ldap_search_attribute string	Atributo de pesquisa LDAP.	
ldap_bind_dn string	DN de ligação LDAP.	
ldap_case_sensitive inteiro	Login com distinção entre maiúsculas e minúsculas do LDAP.	 Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.
ldap_bind_password string	Senha de ligação LDAP.	
saml_auth_enabled integer	Ativar autenticação SAML.	 Valores possíveis: 0 - (padrão) Desativar; 1 - Ativar.
saml_idp_entityid string	ID da entidade IdP SAML.	
saml_sso_url string	URL do serviço SAML SSO.	
saml_slo_url string	URL do serviço SAML SLO.	
saml_username_attribute string	Atributo de nome de usuário SAML.	
saml_sp_entityid string	ID da entidade SAML SP.	
saml_nameid_format string	Formato de ID de nome SAML SP.	
saml_sign_messages inteiro	Assinar mensagens SAML.	 Valores possíveis: 0 - (padrão) Não assinar mensagens; 1 - Assinar mensagens.
saml_sign_assertions integer	SAML assina asserções.	 Valores possíveis: 0 - (padrão) Não assina asserções; 1 - Assine asserções.
saml_sign_authn_requests integer	SAML assina solicitações AuthN.	 Valores possíveis: 0 - (padrão) Não assina solicitações AuthN; 1 - Assina solicitações AuthN.
saml_sign_logout_requests integer	SAML assina solicitações de logout.	 Valores possíveis: 0 - (padrão) Não assina solicitações de logout; 1 - Assina solicitações de logout.
saml_sign_logout_responses integer	SAML assina respostas de logout.	 Valores possíveis: 0 - (padrão) Não assina respostas de logout; 1 - Assina respostas de logout.
saml_encrypt_nameid integer	SAML criptografar ID do nome.	 Valores possíveis: 0 - (padrão) Não criptografar ID do nome; 1 - Criptografar ID do nome.
saml_encrypt_assertions integer	SAML criptografa asserções.	 Valores possíveis: 0 - (padrão) Não criptografa asserções; 1 - Criptografa asserções.
saml_case_sensitive inteiro	Login com distinção entre maiúsculas e minúsculas SAML.	 Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.
passwd_min_length integer	Requisito de comprimento mínimo de senha.	 Intervalo possível de valores: 1-70 8 - default
passwd_check_rules integer	Regras de verificação de senha.	 Os valores de bitmap possíveis são: 0 - verificar o comprimento da senha; 1 - verificar se a senha usa letras latinas maiúsculas e minúsculas; 2 - verifica se a senha usa dígitos; 4 - verifica se a senha usa caracteres especiais; 8 - (padrão) verifica se a senha não está na lista de senhas comumente usadas, não contém derivações de palavra "Zabbix" ou nome de usuário, sobrenome ou nome de usuário.

#####autenticação.atualização {#manual-api-reference-authentication-update}

Descrição

autenticação de objeto.update(autenticação de objeto)

Este método permite atualizar as configurações de autenticação existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Propriedades de autenticação a serem atualizadas.

Valores de retorno

(array) Retorna o array com os nomes dos parâmetros atualizados.

Exemplos

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "authentication.update",
  "params": {
    "http_auth_enabled": 1,
    "http_case_sensitive": 0,
    "http_login_form": 1
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    "http_auth_enabled",
    "http_case_sensitive",
    "http_login_form"
  ],
  "id": 1
}
```

Fonte

CAuthentication::update() em ui/include/classes/api/services/CAuthentication.php.

#####autenticação.get {#manual-api-reference-authentication-get}

Descrição

object authentication.get(object parameters)

O método permite recuperar o objeto de autenticação de acordo com o parâmetros dados.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta apenas um parâmetro.

[Parâmetro|Tipo|Descrição| |-----|-----|-----|-----| |output|query|Este parâmetro é comum para todos os métodos get descritos no [comentário de referência](#).|

Valores de retorno

(object) Retorna o objeto de autenticação.

Exemplos

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "authentication.get",
  "params": {
    "output": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "authentication_type": "0",
    "http_auth_enabled": "0",
    "http_login_form": "0",
    "http_strip_domains": "",
    "http_case_sensitive": "1",
    "ldap_configured": "0",
    "ldap_host": "",
    "ldap_port": "389",
    "ldap_base_dn": "",
    "ldap_search_attribute": "",
    "ldap_bind_dn": "",
    "ldap_case_sensitive": "1",
    "ldap_bind_password": "",
    "saml_auth_enabled": "0",
    "saml_idp_entityid": "",
    "saml_sso_url": "",
    "saml_slo_url": "",
    "saml_username_attribute": "",
    "saml_sp_entityid": "",
    "saml_nameid_format": "",
    "saml_sign_messages": "0",
    "saml_sign_assertions": "0",
    "saml_sign_authn_requests": "0",
    "saml_sign_logout_requests": "0",
    "saml_sign_logout_responses": "0",
    "saml_encrypt_nameid": "0",
    "saml_encrypt_assertions": "0",
    "saml_case_sensitive": "0",
    "passwd_min_length": "8",
    "passwd_check_rules": "8"
  },
  "id": 1
}
```

Fonte

CAuthentication::get() em ui/include/classes/api/services/CAuthentication.php.

Autoregistration

Essa classe foi projetada para trabalhar com o registro automático.

Objetos de referência:

- [Autoregistration](#)

Métodos disponíveis:

- [autoregistration.get](#) - recupera registro automático
- [autoregistration.update](#) - atualiza registro automático

> Objeto de registro automático

Os objetos a seguir estão diretamente relacionados ao autoregistration API.

Registro automático

O objeto de registro automático tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tls_accept integer	Tipo de conexões de entrada permitidas para registro automático.	Valores possíveis: 1 - permitir conexões não seguras; 2 - permitir

TLS com PSK.
3 - permitir ambos inseguro e TLS com conexões PSK.| |tls_psk_identity|string|(somente gravação) String de identidade PSK.
Não coloque informações confidenciais na identidade PSK, elas são transmitidas sem criptografia pela rede para informar a um receptor qual PSK usar.| |tls_psk|string|(somente gravação) string de valor PSK (um número par de caracteres hexadecimais).|

autoregistration.get

Descrição

`object autoregistration.get(object parameters)`

O método permite recuperar o objeto de autoregistro de acordo com o parâmetros dados.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta apenas um parâmetro.

[Parâmetro|Tipo|Descrição| |-----|-----|-----|-----| |output|query|Este parâmetro é comum para todos os métodos get descritos no [comentário de referência](#).|

Valores de retorno

(object) Retorna o objeto de registro automático.

Exemplos

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "autoregistration.get",
  "params": {
    "output": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tls_accept": "3"
  },
  "id": 1
}
```

Fonte

CAutoregistration::get() in ui/include/classes/api/services/CAutoregistration.php.

autoregistration.update

Descrição

`registro automático de objeto.update(registro automático de objeto)`

Este método permite atualizar o autoregistro existente.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Propriedades de autoregistro a serem atualizadas.

Valores de retorno

(boolean) Retorna boolean true como resultado de uma atualização bem-sucedida.

Exemplos**Requisição:**

```
{
  "jsonrpc": "2.0",
  "method": "autoregistration.update",
  "params": {
    "tls_accept": "3",
    "tls_psk_identity": "PSK 001",
    "tls_psk": "11111595725ac58dd977beef14b97461a7c1045b9a1c923453302c5473193478"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": true,
  "id": 1
}
```

Fonte

CAutoregistration::update() em ui/include/classes/api/services/CAutoregistration.php.

Ação

Esta classe foi projetada para trabalhar com ações.

Referências de objetos:

- [Ação](#)
- [Ação condição](#)
- [Ação operação](#)

Métodos disponíveis:

- [action.create](#) - criar novo ações
- [action.delete](#) - delete ações
- [action.get](#) - recupera ações
- [action.update](#) - atualização ações

> Objeto de ação

Os objetos a seguir estão diretamente relacionados à API action

Ação

O objeto de ação tem as seguintes propriedades.

Propriedade	Type	Descrição
actionid	string	(somente leitura) ID da ação.
esc_period (obrigatório)	string	Duração padrão da etapa de operação. Deve ser pelo menos 60 segundos. Aceita segundos, unidade de tempo com sufixo e macro do usuário.
eventsource (obrigatório)	inteiro	Observe que os escalonamentos são suportados apenas para ações triggers, internas e de serviço, e apenas em operações normais. (constante) Tipo de eventos que a ação irá lidar.
nome (obrigatório)	string	Consulte event "source" property para obter uma lista de tipos de eventos suportados. Nome da ação.
status	integer	Se a ação está habilitada ou desabilitada.
pause_suppressed	inteiro	Valores possíveis: 0 - (padrão) habilitado; 1 - desabilitado. Se deve pausar o escalonamento durante os períodos de manutenção ou não.
		Valores possíveis: 0 - Não pausar escalonamento; 1 - (padrão) Pausar escalonamento.
notify_if_canceled	inteiro	Observe que este parâmetro é válido apenas para ações trigger. Se deve notificar quando o escalonamento é cancelado.
		Valores possíveis: 0 - Não notificar quando a escalonamento é cancelado; 1 - (padrão) Notificar quando o escalonamento é cancelado.
		Observe que este parâmetro é válido apenas para ações trigger.

Observe que para alguns métodos (atualizar, excluir), a combinação de parâmetros obrigatórios/opcionais é diferente.

Operação de ação

O objeto de operação de ação define uma operação que será executada quando uma ação é executada. Tem as seguintes propriedades.

[Propriedade|Tipo|Descrição] |-----|-----| |operationid|string|(readonly) ID da operação de ação. | **operationtype**
(obrigatório)|inteiro|Tipo de operação.

Valores possíveis:
0 - enviar mensagem;
1 - script global;
2 - adicionar host;
3 - remover host;
4 - adicionar ao grupo de hosts;
5 - remover do grupo de hosts;
6 - vincular ao modelo;
7 - desvincular do modelo;
 >8 - habilita o host;
9 - desabilita o host;
10 - define o modo de inventário do host.

Observe que apenas os tipos '0' e '1' são suportados para ações de acionador e serviço, apenas '0' é suportado para ações internas. Todos os tipos têm suporte para ações de descoberta e registro automático. | actionid|string|(readonly) ID da ação à qual a operação pertence. | esc_period|string|Duração de uma etapa de escalação em segundos. Deve ser superior a 60 segundos. Aceita segundos, unidade de tempo com sufixo e macro de usuário. Se definido como 0 ou 0s, o período de escalonamento de ação padrão será usado.

Padrão: 0s.

Observe que os escalonamentos são suportados apenas para ações de gatilho, internas e de serviço, e apenas em condições normais operações. | esc_step_from|integer|Etapa para iniciar o escalonamento.

Padrão: 1.

Observe que os escalonamentos são suportados apenas para ações de gatilho, internas e de serviço, e apenas em condições normais operações. | esc_step_to|integer|Passo para encerrar o escalonamento em.

Padrão: 1.

Observe que os escalonamentos são suportados apenas para ações de gatilho, internas e de serviço, e apenas em operações normais . | evaltype|integer|Método de avaliação da condição de operação.

Valores possíveis:
0 - (padrão) AND / OR;
1 - AND;
2 - OR. | opcommand|object|Objeto contendo dados no script global executado pela operação.

Cada objeto tem uma propriedade a seguir: scriptid - (string) ID do script.

Obrigatório para operações globais de script. | opcommand_grp|array|Grupos de hosts para executar scripts globais.

Cada objeto tem as seguintes propriedades:
opcommand_grpid - (string, readonly) ID do objeto;
 operationid - (string, readonly) ID da operação;
groupid - (string) ID do grupo de hosts.

Obrigatório para operações globais de script se opcommand_hst for não definido. | opcommand_hst|array|Host para executar scripts globais.

Cada objeto tem as seguintes propriedades:
opcommand_hstid - (string, read-

only) ID do objeto;
operationid - (string, readonly) ID da operação;
hostid - (string) ID do host; se definido como 0, o comando será executado no host atual.

Requerido para operações globais de script se opcommand_grp não estiver definido.| |opconditions|array|Condições de operação usadas para ações de gatilho.

O objeto de condição de operação é **descrito em detalhes abaixo**.| |opgroup|array|Grupos de hosts aos quais adicionar hosts.

Cada objeto tem as seguintes propriedades:
operationid - (string, readonly) ID da operação;
groupid - (string) ID do grupo de hosts.

Obrigatório para as operações "adicionar ao grupo de hosts" e "remover do grupo de hosts".| |opmessage|object|Objeto contendo os dados sobre a mensagem enviada pela operação.

O objeto mensagem da operação é **descrito em detalhes abaixo**.

Requerido para operações de mensagens.| |opmessage_grp|array|Grupos de usuários para os quais enviar mensagens.

Cada objeto tem as seguintes propriedades:
operationid - (string, readonly) ID da operação;
usrgrp - (string) ID do grupo de usuários.

Requerido para operações de mensagem se opmessage_usr não estiver definido.| |opmessage_usr|array|Usuários para quem enviar mensagens.

Cada objeto tem as seguintes propriedades:
operationid - (string, readonly) ID da operação;
userid - (string) ID do usuário.

Obrigatório para operações de mensagem se opmessage_grp não estiver definido.| |optemplate|array|Modelos aos quais vincular os hosts.

Cada objeto tem as seguintes propriedades:
operationid - (string, readonly) ID da operação;
templateid - (string) ID do modelo.

Obrigatório para as operações "vincular ao modelo" e "desvincular do modelo".| |opinventory|object|Modo de inventário definido como host.

O objeto tem as seguintes propriedades:
operationid - (string, readonly) ID da operação;
inventory_mode - * (string)* Modo de inventário.

Obrigatório para as operações "Definir modo de inventário do host".|

Mensagem de operação de ação

O objeto de mensagem de operação contém dados sobre a mensagem que será enviada pela operação.

|Propriedade|**Tipo**|Descrição| |-----|-----| |default_msg|integer|Se usar o texto e o assunto da mensagem de ação padrão.

Valores possíveis:
0 - usar os dados da operação;
1 - (padrão) usar os dados do tipo de mídia.| |mediatypeid|string|ID do tipo de mídia que será usado para enviar a mensagem.| |message|string|Texto da mensagem da operação.| |assunto|string|Assunto da mensagem da operação.|

Condição de operação da ação

O objeto de condição de operação de ação define uma condição que deve ser atendida para realizar a operação atual. Tem as seguintes propriedades.

|Propriedade|**Tipo**|Descrição| |-----|-----| |opconditionid|string|(readonly) ID da condição de operação da ação| |**conditiontype**
(obrigatório)|inteiro|Tipo de condição.

Valores possíveis:
14 - evento reconhecido.| |**valor**
(obrigatório)|string|Valor para comparar.| |operationid|string|(readonly) ID da operação.| |operador|inteiro|Operador de condição.

Valores possíveis:
0 - (padrão) =.|

Os seguintes operadores e valores são suportados para cada operação tipo de condição.

|Condição|Nome da condição|Operadores suportados|Valor esperado| |-----|-----|-----| |14|Evento reconhecido|=|Se o evento é reconhecido.

Valores possíveis:
0 - não reconhecido;
1 - reconhecido.|

Operação de recuperação

O objeto de operação de recuperação define uma operação que será realizada quando um problema for resolvido. Operações de recuperação são possíveis para ações de trigger, internas e de serviço. Ele possui as seguintes propriedades.

Propriedade	Type	Descrição
operationid	string	(somente leitura) ID da operação da ação.
operationtype (obrigatório)	integer	Tipo da operação. Valores possíveis para ações de trigger e serviço: 0 - enviar mensagem; 1 - script global; 11 - notificar todos envolvidos. Valores possíveis para ações internas: 0 - enviar mensagem; 11 - notificar todos envolvidos.
actionid	string	(somente leitura) ID da ação à qual a operação de recuperação pertence.

|opcommand_hst|array|Host para executar scripts globais.

Cada objeto tem as seguintes propriedades:
hostid - (string) ID do host; se definido como 0, o comando será executado no host atual.

Requerido para operações globais de script se opcommand_grp não estiver definido.| |opmessage|object|Objeto contendo os dados sobre a mensagem enviada pela operação de atualização.

O objeto de mensagem da operação é **descrito em detalhes acima**.| |opmessage_grp|array|Grupos de usuários para os quais enviar mensagens.

Cada objeto tem as seguintes propriedades:
usrgrp_id - (string) ID do grupo de usuários.

Requerido apenas para operações send message se opmessage_usr não estiver definido.
É ignorado para operações send update message.| |opmessage_usr|array|Usuários para os quais enviar mensagens.

Cada objeto tem as seguintes propriedades:
user_id - (string) ID do usuário.

Obrigatório somente para operações send message se opmessage_grp não estiver definido.
É ignorado para operações send update message.|

Filtro de ação

O filtro de ação define um conjunto de condições que devem ser atendidas para executar as operações de ação configuradas. Tem o seguinte propriedades.

|Propriedade|Tipo|Descrição| |-----|-----| |condições
(obrigatório)|array|Conjunto de condições de filtro a serem usadas para filtrar resultados.| |evaltype
(obrigatório)|inteiro|Método de avaliação da condição do filtro.

Valores possíveis:
0 - e/ou;
1 - e;
2 - ou;
3 - expressão personalizada.| |eval_formula|string|(readonly) Expressão gerada que será utilizada para avaliar as condições do filtro. A expressão contém IDs que fazem referência a condições de filtro específicas por seu formulaid. O valor de eval_formula é igual ao valor de formula para filtros com uma expressão personalizada.| |formula|string|Expressão definida pelo usuário a ser usada para avaliar condições de filtros com uma expressão customizada. A expressão deve conter IDs que façam referência a condições de filtro específicas por seu formulaid. Os IDs usados na expressão devem corresponder exatamente aos definidos nas condições do filtro: nenhuma condição pode permanecer sem uso ou omitida.

Obrigatório para filtros de expressão personalizados.|

Condição do filtro de ação

O filtro de ação define uma condição específica que deve ser verificada antes de executar as operações de ação.

|Propriedade|Tipo|Descrição| |-----|-----| |conditionid|string|(readonly) ID da condição da ação.| |conditiontype
(obrigatório)|inteiro|Tipo de condição.

Valores possíveis para ações de gatilho:
0 - grupo de hosts;
1 - host;
2 - gatilho;
3 - nome do gatilho;
4 - gravidade do gatilho;
6 - período de tempo;
13 - modelo de host;
16 - problema suprimido;
25 - evento tag;
26 - valor da tag de evento.

Valores possíveis para ações de descoberta:
7 - IP do host;
8 - tipo de serviço descoberto;
9 - porta de serviço descoberto;
10 - status da descoberta;
11 - duração do tempo de atividade ou do tempo de inatividade;
12 - valor recebido;
18 - regra de descoberta;
19 - verificação de descoberta;
20 - proxy;
21 - objeto de descoberta.

Valores possíveis para ações de registro automático:
20 - proxy;
22 - nome do host;
24 - metadados do host.

Valores possíveis para ações internas:
0 - grupo de host;
1 - host;
13 - modelo de host;
23 - tipo de evento;
25 - tag de evento;
26 - valor da tag de evento .

Valores possíveis para ações de serviço:
25 - tag de evento;
26 - valor da tag de evento;
27 - serviço;
28 - nome do serviço.| |valor
(obrigatório)|string|Valor para comparar.| |value2
|string|Valor secundário para comparar. Necessário para ações de gatilho, internas e de serviço quando o tipo de condição é 26.| |actionid|string|(readonly) ID da ação à qual a condição pertence.| |formulaid|string|ID exclusivo arbitrário que é usado para fazer referência à condição de uma expressão customizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente.| |operator|inteiro|Operador de condição.

Valores possíveis:
0 - (padrão) é igual;
1 - não é igual;
2 - contém;
3 - não contém;
4 - em;
5 - é maior ou igual;
6 - é menor ou igual;
7 - não em;
8 - corresponde;
9 - não corresponde;
10 - Sim;
11 - Não.|

Para entender melhor como usar filtros com vários tipos de expressões, veja exemplos no **action.get** e método **action.create** Páginas.

Os seguintes operadores e valores são suportados para cada condição tipo.

|Condição|Nome da condição|Operadores suportados|Valor esperado| |-----|-----| |0|Grupo de hosts|igual,
diferente|ID do grupo de hosts.| |1|Host|igual,
não é igual|Host ID.| |2|Trigger|igual,
diferente|Trigger ID.| |3|Nome do gatilho|contém,
não contém|Nome do gatilho.| |4|Gravidade do gatilho|igual,
diferente,
é maior ou igual,
é menor ou igual|Gravidade do gatilho. Consulte a **propriedade "severity" do gatilho** para obter uma lista de gravidades de gatilho compatíveis.| |5|Valor do gatilho|igual|Valor do gatilho. Consulte a **propriedade "valor" do gatilho** para obter uma lista de valores de gatilho compatíveis.| |6|Período de tempo|em, não em|Tempo em que o evento foi acionado como um **período de tempo**.| |7|IP do host|igual,
não é igual|Um ou vários intervalos de IP para verificar separados por vírgulas. Consulte a seção **configuração de descoberta de rede** para obter mais informações sobre os formatos suportados de intervalos de IP.| |8|Tipo de serviço descoberto|igual,
não é igual|Tipo de serviço descoberto. O tipo de serviço corresponde ao tipo de verificação de descoberta usado para detectar o serviço. Consulte a **propriedade de "tipo" de verificação de descoberta** para obter uma lista de tipos suportados.| |9|Porta de serviço descoberto|igual,
não é igual|Um ou vários intervalos de portas separados por vírgulas.| |10|Status de descoberta|igual|Status de um objeto descoberto.

Valores possíveis:
0 - host ou serviço ativo;
1 - host ou serviço desativado;
2 - host ou serviço descoberto;
3 - host ou serviço perdido.| |11|A du

ração do tempo de atividade ou inatividade|é maior ou igual,
é menor ou igual|Tempo que indica há quanto tempo o objeto descoberto está no status atual em segundos.| |12|Valores recebidos|igual,
diferente,
é maior ou igual,
é menor ou igual,
contém,
não contém|Valor retornado ao realizar um Verificação de descoberta do agente Zabbix, SNMPv1, SNMPv2 ou SNMPv3.| |13|Modelo do host|igual,
não é igual|ID do modelo vinculado.| |16|O problema foi suprimido|Sim, Não|Nenhum valor necessário: usar o operador "Sim" significa que o problema deve ser suprimido, "Não" - não suprimido.| |18|Regra de descoberta|igual,
não é igual|ID da regra de descoberta.| |19|Verificação de descoberta|igual,
não é igual|ID da verificação de descoberta.| |20|Proxy|igual,
diferente|ID do proxy.| |21|Objeto de descoberta|igual|Tipo de objeto que acionou o evento de descoberta.

Valores possíveis:
1 - host descoberto;
2 - serviço descoberto.| |22|Nome do host|contém,
não contém,
corresponde,
não corresponde|Nome do host.
O uso de uma expressão regular é compatível com os operadores corresponde e não corresponde em condições de registro automático.| |23|Tipo de evento|igual|Evento interno específico.

Valores possíveis:
0 - item no estado "não suportado";
1 - item no estado "normal";
2 - Regra LLD em estado "não suportado";
3 - Regra LLD em estado "normal";
4 - acionador em estado "desconhecido";
5 - acionador em estado "normal".| |24|Metadados do host|contém,
não contém,
corresponde,
não corresponde|Metadados do host registrado automaticamente.
O uso de uma expressão regular é compatível com os operadores corresponde e não corresponde.| |25|Tag|igual,
diferente,
contém,
não contém|Etiqueta de evento.| |26|Valor da tag|igual,
não é igual,
contém,
não contém|Valor da tag do evento.| |27|Serviço|é igual,
não é igual a|ID do serviço.| |28|Nome do serviço|igual,
não é igual|Nome do serviço.|

action.update

Descrição

`object action.update(object/array actions)`

Este método permite atualizar as ações existentes.

Note:

Este método está disponível apenas para usuários do tipo Admin e Super admin. As permissões para chamar o método podem ser revogadas na configuração das funções do usuário. Veja [Funções do usuário] /manual/web_interface/frontend_sections/administration/user_roles) para mais informações.

Parâmetros

(object/array) Propriedades da ação a serem atualizadas.

A propriedade `actionid` deve ser definida para cada ação, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das **propriedades de ação padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
<code>filter</code>	object	Action filter objeto para substituir o filtro atual.
<code>operações</code>	array	Ação operações para substituir as operações existentes.
<code>recovery_operations</code>	array	Ação operações de recuperação para substituir as operações de recuperação existentes.
<code>update_operations</code>	array	Ação operações de atualização para substituir as operações de atualização existentes.

Valores de retorno

(object) Retorna um objeto contendo os IDs das ações atualizadas sob a propriedade `actionids`.

Exemplos

Desativar ação

Desabilite a ação, ou seja, defina seu status como "1".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "action.update",
  "params": {
    "actionid": "2",
    "status": "1"
  }
}
```

```

    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
  }
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "actionids": [
      "2"
    ]
  },
  "id": 1
}

```

Veja também

- [filtro de ação](#)
- [Operação de ação](#)

Fonte

CAction::update() em ui/include/classes/api/services/CAction.php.

#####ação.criar {#manual-api-reference-action-create}

COMENTÁRIO <https://git.zabbix.com/projects/WEB/repos/documentation/browse/en/manual/api/reference/action/create.md?at=refs%2Fheads%2Fmaster>

CONTEXTO c7f202d4 RECURSO Documentação 6.0•manual/api/reference/action/create.xliff

Descrição

object action.create(object/array actions)

Este método permite criar novas ações.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(objeto/array) Ações para criar.

Além das [propriedades de ação padrão](#), o O método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição| |-----|-----|-----| |filter|objeto|Ação [filtro](#) objeto para a ação.| |oper-
ações|array|Ação [operações](#) para criar para a ação.| |recovery_operations|array|Ação [operações de recuperação](#) para criar para a
ação.| |update_operations|array|Ação [update operations](#) para criar para a ação.]

Valores de retorno

(object) Retorna um objeto contendo os IDs das ações criadas sob a propriedade actionids. A ordem dos IDs retornados corresponde a ordem das ações passadas.

Exemplos

Crie uma ação trigger

Criar uma ação de gatilho que será acionada quando um gatilho (com a palavra "memória" em seu nome) do host "10084" entrar em estado de PROBLEMA. A ação terá 4 operações configuradas. A primeira operação imediata enviará uma mensagem para todos os usuários no grupo de usuários "7" via tipo de mídia "1". Se o evento não for resolvido em 30 minutos, a segunda operação executará o [script](#) "5" (script com escopo "Operação de ação") em todos os hosts do grupo "2". Se o evento for resolvido, uma operação de recuperação notificará todos os usuários que receberam quaisquer mensagens relacionadas ao problema. Se o evento for atualizado, uma operação de reconhecimento/atualização notificará (com um assunto e mensagem personalizados) todos os usuários que receberam quaisquer mensagens relacionadas ao problema.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "method": "action.create",

```

```

"params": {
  "name": "Trigger action",
  "eventsourcesource": 0,
  "esc_period": "30m",
  "filter": {
    "evaltype": 0,
    "conditions": [
      {
        "conditiontype": 1,
        "operator": 0,
        "value": "10084"
      },
      {
        "conditiontype": 3,
        "operator": 2,
        "value": "memory"
      }
    ]
  },
  "operations": [
    {
      "operationtype": 0,
      "esc_step_from": 1,
      "esc_step_to": 1,
      "opmessage_grp": [
        {
          "usrgrp": "7"
        }
      ],
      "opmessage": {
        "default_msg": 1,
        "mediatypeid": "1"
      }
    },
    {
      "operationtype": 1,
      "esc_step_from": 2,
      "esc_step_to": 2,
      "opconditions": [
        {
          "conditiontype": 14,
          "operator": 0,
          "value": "0"
        }
      ],
      "opcommand_grp": [
        {
          "groupid": "2"
        }
      ],
      "opcommand": {
        "scriptid": "5"
      }
    }
  ],
  "recovery_operations": [
    {
      "operationtype": "11",
      "opmessage": {
        "default_msg": 1
      }
    }
  ]
}

```

```

    ],
    "update_operations": [
      {
        "operationtype": "12",
        "opmessage": {
          "default_msg": 0,
          "message": "Custom update operation message body",
          "subject": "Custom update operation message subject"
        }
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "actionids": [
      "17"
    ]
  },
  "id": 1
}

```

Crie uma ação de descoberta

Crie uma ação de descoberta que vinculará os hosts descobertos ao template "10091".

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "ação.criar",
  "parâmetros": {
    "name": "Ação de descoberta",
    "origem do evento": 1,
    "estado": 0,
    "filtro": {
      "evaltype": 0,
      "condições": [
        {
          "tipo de condição": 21,
          "operador": 0,
          "valor": "1"
        },
        {
          "tipo de condição": 10,
          "operador": 0,
          "valor": "2"
        }
      ]
    }
  },
  "operações": [
    {
      "tipo de operação": 6,
      "otimizar": [
        {
          "templateid": "10091"
        }
      ]
    }
  ]
}

```

```

    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "actionids": [
      "18"
    ]
  },
  "id": 1
}

```

Usando um filtro de expressão personalizada

Crie uma ação trigger que utiliza uma expressão personalizada - "A e (B ou C)" - para avaliar as condições da ação. Uma vez que um gatilho com uma severidade maior ou igual a "Aviso" do host "10084" ou do host "10106" entra em estado de PROBLEMA, a ação enviará uma mensagem para todos os usuários no grupo de usuários "7" via tipo de mídia "1". Os IDs das fórmulas "A", "B" e "C" foram escolhidos arbitrariamente.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "method": "action.create",
  "params": {
    "name": "Trigger action",
    "eventsources": 0,
    "esc_period": "15m",
    "filter": {
      "evaltype": 3,
      "formula": "A and (B or C)",
      "conditions": [
        {
          "conditiontype": 4,
          "operator": 5,
          "value": "2",
          "formulaid": "A"
        },
        {
          "conditiontype": 1,
          "operator": 0,
          "value": "10084",
          "formulaid": "B"
        },
        {
          "conditiontype": 1,
          "operator": 0,
          "value": "10106",
          "formulaid": "C"
        }
      ]
    },
    "operations": [
      {
        "operationtype": 0,
        "esc_step_from": 1,
        "esc_step_to": 1,
        "opmessage_grp": [

```



```
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "actionids": [
      19
    ]
  },
  "id": 1
}
```

Consulte também

- [Action filter](#)
- [Action operation](#)
- [Script](#)

Fonte

CAction::create() in ui/include/classes/api/services/CAction.php.

ação.excluir

Descrição

`object action.delete(array actionIds)`

Este método permite excluir ações.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs das ações a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das ações excluídas sob a propriedade `actionids`.

Exemplos

Excluir várias ações

Excluir duas ações.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "action.delete",
  "params": [
    "17",
    "18"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
```

```

        "actionids": [
            "17",
            "18"
        ],
    },
    "id": 1
}

```

Fonte

CAction::delete() in ui/include/classes/api/services/CAction.php.

#####ação.obter {#manual-api-reference-action-get}

Descrição

integer/array action.get(object parameters)

O método permite recuperar ações de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----|-----| |actionids|string/array|Retorna apenas ações com os IDs fornecidos.| |groupids|string/array|Retorna apenas as ações que usam os grupos de hosts fornecidos em condições de ação.| |hostids|string/array|Retorna apenas ações que usam os hosts fornecidos em condições de ação.| |triggerids|string/array|Retorna apenas ações que usam os gatilhos fornecidos em condições de ação.| |mediatypeids|string/array|Retorna apenas ações que usam os tipos de mídia fornecidos para enviar mensagens.| |usrgrpids|string/array|Retorna apenas as ações configuradas para enviar mensagens para os grupos de usuários especificados.| |userids|string/array|Retorna apenas as ações configuradas para enviar mensagens para os usuários especificados.| |scriptids|string/array|Retorna apenas as ações configuradas para executar os scripts fornecidos.| |selectFilter|query|Retorne uma propriedade **filter** com o filtro de condição de ação.| |selectOperations|query|Retorne uma propriedade **operações** com operações de ação.| |selectRecoveryOperations|query|Retorne uma propriedade **recovery_operations** com operações de recuperação de ação.| |selectUpdateOperations|query|Retorne uma propriedade **update_operations** com operações de atualização de ação.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: actionid, name e status.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos no [comentário de referência](#).| |editável|boolean|^| |excludePesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna ou:

- uma array de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar ações de trigger

Recupere todas as ações de trigger configuradas junto com as condições de ação e operações.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {
        "output": "extend",
        "selectOperations": "extend",
        "selectRecoveryOperations": "extend",
        "selectUpdateOperations": "extend",
    }
}

```

```

    "selectFilter": "extend",
    "filter": {
        "eventsources": 0
    }
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "3",
            "name": "Report problems to Zabbix administrators",
            "eventsources": "0",
            "status": "1",
            "esc_period": "1h",
            "pause_suppressed": "1",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [],
                "eval_formula": ""
            },
            "operations": [
                {
                    "operationid": "3",
                    "actionid": "3",
                    "operationtype": "0",
                    "esc_period": "0",
                    "esc_step_from": "1",
                    "esc_step_to": "1",
                    "evaltype": "0",
                    "opconditions": [],
                    "opmessage": [
                        {
                            "default_msg": "1",
                            "subject": "",
                            "message": "",
                            "mediatypeid" => "0"
                        }
                    ],
                    "opmessage_grp": [
                        {
                            "usrgrpid": "7"
                        }
                    ]
                }
            ],
            "recovery_operations": [
                {
                    "operationid": "7",
                    "actionid": "3",
                    "operationtype": "11",
                    "evaltype": "0",
                    "opconditions": [],
                    "opmessage": {
                        "default_msg": "0",
                        "subject": "{TRIGGER.STATUS}: {TRIGGER.NAME}",
                        "message": "Trigger: {TRIGGER.NAME}\r\nTrigger status: {TRIGGER.STATUS}\r\nTrigger"
                    }
                }
            ]
        }
    ]
}

```

```

        "mediatypeid": "0"
    }
}
],
"update_operations": [
    {
        "operationid": "31",
        "operationtype": "12",
        "evaltype": "0",
        "opmessage": {
            "default_msg": "1",
            "subject": "",
            "message": "",
            "mediatypeid": "0"
        }
    },
    {
        "operationid": "32",
        "operationtype": "0",
        "evaltype": "0",
        "opmessage": {
            "default_msg": "0",
            "subject": "Updated: {TRIGGER.NAME}",
            "message": "{USER.FULLNAME} updated problem at {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}",
            "mediatypeid": "1"
        }
    },
    "opmessage_grp": [
        {
            "usrgrp": {
                "usrgrp": "7"
            }
        }
    ],
    "opmessage_usr": [],
},
{
    "operationid": "33",
    "operationtype": "1",
    "evaltype": "0",
    "opcommand": {
        "scriptid": "3"
    },
    "opcommand_hst": [
        {
            "hostid": "10084"
        }
    ],
    "opcommand_grp": []
}
]
}
],
"id": 1
}

```

Recuperar ações de descoberta

Recupere todas as ações de descoberta configuradas junto com a ação condições e operações. O filtro usa o tipo de avaliação "and" evaluation type, então a propriedade formula está vazia e eval_formula é gerado automaticamente.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "action.get",
    "params": {

```

```

        "output": "extend",
        "selectOperations": "extend"
        "selectFilter": "extend",
        "filter": {
            "eventsources": 1
        }
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": [
        {
            "actionid": "2",
            "name": "Auto discovery. Linux servers.",
            "eventsources": "1",
            "status": "1",
            "esc_period": "0s",
            "pause_suppressed": "1",
            "filter": {
                "evaltype": "0",
                "formula": "",
                "conditions": [
                    {
                        "conditiontype": "10",
                        "operator": "0",
                        "value": "0",
                        "value2": "",
                        "formulaid": "B"
                    },
                    {
                        "conditiontype": "8",
                        "operator": "0",
                        "value": "9",
                        "value2": "",
                        "formulaid": "C"
                    },
                    {
                        "conditiontype": "12",
                        "operator": "2",
                        "value": "Linux",
                        "value2": "",
                        "formulaid": "A"
                    }
                ]
            },
            "eval_formula": "A and B and C"
        },
        {
            "operations": [
                {
                    "operationid": "1",
                    "actionid": "2",
                    "operationtype": "6",
                    "esc_period": "0s",
                    "esc_step_from": "1",
                    "esc_step_to": "1",
                    "evaltype": "0",
                    "opconditions": [],
                    "optemplate": [

```

```

        "templateid": "10001"
    }
    ],
    {
        "operationid": "2",
        "actionid": "2",
        "operationtype": "4",
        "esc_period": "0s",
        "esc_step_from": "1",
        "esc_step_to": "1",
        "evaltype": "0",
        "opconditions": [],
        "opgroup": [
            {
                "groupid": "2"
            }
        ]
    }
    ],
    {
        "id": 1
    }
}

```

Veja também

- [Action filter](#)
- [Action operation](#)

Fonte

CAction::get() in ui/include/classes/api/services/CAction.php.

Cenário Web

Esta classe foi projetada para trabalhar com cenários web.

Referências de objetos:

- [Web scenario](#)
- [Scenario step](#)

Métodos disponíveis:

- [httptest.create](#) - criando novos cenários web
- [httptest.delete](#) - deletando cenários web
- [httptest.get](#) - recuperando cenários web
- [httptest.update](#) - atualizando cenários web

> Objeto de cenário Web

Os objetos a seguir estão diretamente relacionados à API webcheck.

Cenário Web

O objeto de cenário da web tem as seguintes propriedades.

Propriedade	Tipo	Descrição
httptestid	string	(somente leitura) ID do cenário web.
hostid (requirido)	string	ID do host ao qual o cenário da web pertence.
name (requirido)	string	Nome do cenário web.
agent	string	String do agente do usuário que será usada pelo cenário web.
authentication	integer	<p>Padrão: Zabbix</p> <p>Método de autenticação que será usado pelo cenário web.</p> <p>Valores possíveis:</p> <p>0 - (padrão) none;</p> <p>1 - autenticação HTTP básica;</p> <p>2 - Autenticação NTLM.</p>
delay	string	Intervalo de execução do cenário web. Aceita segundos, unidade de tempo com sufixo e macro do usuário..
headers	array de HTTP fields	Padrão: 1m. cabeçalhos HTTP que serão enviados ao realizar uma solicitação.
http_password	string	Senha usada para autenticação básica HTTP ou NTLM.
http_proxy	string	Proxy que será usado pelo cenário da web dado como http://[username[:password]@]proxy.example.com[:port].
http_user	string	Nome de usuário usado para autenticação HTTP ou NTLM básica.
nextcheck	timestamp	(somente leitura) Hora da próxima execução do cenário web.
retries	integer	Número de vezes que um cenário da Web tentará executar cada etapa antes de falhar.
ssl_cert_file	string	Padrão: 1. Nome do arquivo de certificado SSL usado para autenticação do cliente (deve estar no formato PEM).
ssl_key_file	string	Nome do arquivo de chave privada SSL usado para autenticação do cliente (deve estar no formato PEM).
ssl_key_password	string	Senha da chave privada SSL.
status	integer	Se o cenário da Web está ativado.
templateid	string	Os valores possíveis são: 0 - (padrão) Ativado; 1 - Desativado. (somente leitura) ID do cenário Web do modelo pai.
variables	array of HTTP fields	Variáveis de cenário web.
verify_host	integer	Se deve verificar se o nome do host especificado no certificado SSL corresponde ao usado no cenário.
verify_peer	integer	<p>Os valores possíveis são:</p> <p>0 - (padrão) pular a verificação do host;</p> <p>1 - verificar host.</p> <p>Se deve verificar o certificado SSL do servidor web.</p>
uuid	string	<p>Os valores possíveis são:</p> <p>0 - (padrão) pular verificação de pares;</p> <p>1 - verificar par.</p> <p>(somente leitura em cenários da web já existentes)</p> <p>Identificador exclusivo global, usado para vincular cenários Web importados a cenários já existentes. Usado apenas para cenários Web em modelos.</p>

Cenário Web

O objeto de cenário da web tem as seguintes propriedades.

Propriedade	Tipo	Descrição
httptestid	string	(somente leitura) ID do cenário web.
hostid (requirido)	string	ID do host ao qual o cenário da web pertence.
name (requirido)	string	Nome do cenário web.
agent	string	String do agente do usuário que será usada pelo cenário web.
authentication	integer	<p>Padrão: Zabbix</p> <p>Método de autenticação que será usado pelo cenário web.</p> <p>Valores possíveis:</p> <p>0 - (padrão) none;</p> <p>1 - autenticação HTTP básica;</p> <p>2 - Autenticação NTLM.</p>
delay	string	Intervalo de execução do cenário web. Aceita segundos, unidade de tempo com sufixo e macro do usuário..
headers	array de HTTP fields	Padrão: 1m. cabeçalhos HTTP que serão enviados ao realizar uma solicitação.
http_password	string	Senha usada para autenticação básica HTTP ou NTLM.
http_proxy	string	Proxy que será usado pelo cenário da web dado como http://[username[:password]@]proxy.example.com[:port].
http_user	string	Nome de usuário usado para autenticação HTTP ou NTLM básica.
nextcheck	timestamp	(somente leitura) Hora da próxima execução do cenário web.
retries	integer	Número de vezes que um cenário da Web tentará executar cada etapa antes de falhar.
ssl_cert_file	string	Padrão: 1. Nome do arquivo de certificado SSL usado para autenticação do cliente (deve estar no formato PEM).
ssl_key_file	string	Nome do arquivo de chave privada SSL usado para autenticação do cliente (deve estar no formato PEM).
ssl_key_password	string	Senha da chave privada SSL.
status	integer	Se o cenário da Web está ativado.
templateid	string	Os valores possíveis são: 0 - (padrão) Ativado; 1 - Desativado. (somente leitura) ID do cenário Web do modelo pai.
variables	array of HTTP fields	Variáveis de cenário web.
verify_host	integer	Se deve verificar se o nome do host especificado no certificado SSL corresponde ao usado no cenário.
verify_peer	integer	<p>Os valores possíveis são:</p> <p>0 - (padrão) pular a verificação do host;</p> <p>1 - verificar host.</p> <p>Se deve verificar o certificado SSL do servidor web.</p>
uuid	string	<p>Os valores possíveis são:</p> <p>0 - (padrão) pular verificação de pares;</p> <p>1 - verificar par.</p> <p>(somente leitura em cenários da web já existentes)</p> <p>Identificador exclusivo global, usado para vincular cenários Web importados a cenários já existentes. Usado apenas para cenários Web em modelos.</p>

Etapa do cenário

O objeto de etapa de cenário define uma verificação de cenário da web específica. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
httpstepid	string	(somente leitura) ID da etapa do cenário.
name (requirido)	string	Nome da etapa do cenário.
no (requirido)	integer	Número de sequência da etapa em um cenário web.
url (requirido)	string	URL a ser verificado.
follow_redirects	integer	Se deve seguir redirecionamentos HTTP. Os valores possíveis são: 0 - não siga redirecionamentos; 1 - (padrão) seguir redirecionamentos.
headers	array de HTTP fields	Cabeçalhos HTTP que serão enviados ao realizar uma solicitação. Os cabeçalhos das etapas do cenário substituirão os cabeçalhos especificados para o cenário web.
httptestid	string	(somente leitura) ID do cenário da web ao qual a etapa pertence.
posts	string array de HTTP fields	Variáveis HTTP POST como uma string (dados brutos de postagem) ou como uma matriz de HTTP fields (dados de campo de formulário).
required	string	Texto que deve estar presente na resposta.
retrieve_mode	integer	Parte da resposta HTTP que a etapa do cenário deve recuperar. Os valores possíveis são: 0 - (padrão) somente body; 1 - somente headers; 2 - headers e body.
status_codes	string	Intervalos de códigos de status HTTP necessários separados por vírgulas.
timeout	string	Solicitar tempo limite em segundos. Aceita segundos, unidade de tempo com sufixo e macro do usuário.
variables	array de HTTP fields	Padrão: 15s. Máximo: 1h. Mínimo: 1s. Variáveis da etapa do cenário.
query_fields	array de HTTP fields	Campos de consulta - array de HTTP fields que será adicionado ao URL ao realizar uma solicitação

Campo HTTP

O objeto de campo HTTP define um nome e um valor que é usado para especificar a variável, o cabeçalho HTTP, os dados do campo de formulário POST dos dados do campo de consulta. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
name (required)	string	Nome do cabeçalho / variável / Campo POST ou GET.
value (required)	string	Valor do cabeçalho / variável / Campo POST ou GET.

httptest.create

Descrição

```
object httptest.create(object/array webScenarios)
```

Este método permite criar novos cenários web.

Note:

A criação de um cenário da web criará automaticamente um conjunto de **itens de monitoramento da web**.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) criar Cenários Web.

Além das **propriedades padrão do cenário da web**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
steps (requerido)	array	Cenário Web steps .
tags	array	Cenário Web tags .

Valores de retorno

(object) Retorna um objeto contendo os IDs dos cenários da web criados na propriedade `httptestids`. A ordem dos IDs retornados corresponde à ordem dos cenários da Web passados.

Exemplos**Criando um cenário web**

Crie um cenário web para monitorar a página inicial da empresa. O cenário terá duas etapas, para verificar a página inicial e a página "About" e garantir que retornem o código de status HTTP 200.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "httptest.create",
  "params": {
    "name": "Homepage check",
    "hostid": "10085",
    "steps": [
      {
        "name": "Homepage",
        "url": "http://example.com",
        "status_codes": "200",
        "no": 1
      },
      {
        "name": "Homepage / About",
        "url": "http://example.com/about",
        "status_codes": "200",
        "no": 2
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "httptestids": [
      "5"
    ]
  },
  "id": 1
}
```

Veja também

- [Scenario step](#)

Fonte

CHttpTest::create() in ui/include/classes/api/services/CHttpTest.php.

httptest.delete

Descrição

`object httptest.delete(array webScenarioIds)`

Este método permite excluir cenários web.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos cenários Web a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos cenários da web excluídos na propriedade `httptestids`.

Exemplos

Excluindo vários cenários web

Excluir dois cenários web.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "httptest.delete",
  "params": [
    "2",
    "3"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "httptestids": [
      "2",
      "3"
    ]
  },
  "id": 1
}
```

Fonte

CHttpTest::delete() in ui/include/classes/api/services/CHttpTest.php.

httptest.get

Descrição

`integer/array httptest.get(object parameters)`

O método permite recuperar cenários da web de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
groupids	string/array	Retornar apenas cenários Web que pertencem aos grupos de hosts fornecidos.
hostids	string/array	Retornar apenas cenários Web que pertencem aos hosts fornecidos.
httpstestids	string/array	Retornar apenas cenários web com os IDs fornecidos.
inherited	boolean	Se definido como true, retorna apenas cenários web herdados de um modelo.
monitored	boolean	Se definido como true, retorna apenas cenários Web habilitados que pertencem a hosts monitorados.
templated	boolean	Se definido como true, retornará apenas cenários da Web que pertencem a modelos.
templateids	string/array	Retornar apenas cenários da Web que pertencem aos modelos fornecidos.
expandName	flag	Expandir macros no nome do cenário web.
expandStepName	flag	Expandir macros nos nomes das etapas do cenário.
evaltype	integer	Regras para pesquisa de tags. Valores possíveis: 0 - (padrão) And/Or; 2 - Or.
tags	array de objetos	Retorne apenas cenários Web com as tags fornecidas. Correspondência exata por tag e pesquisa com distinção entre maiúsculas e minúsculas por valor de tag, dependendo do valor do operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Uma matriz vazia retorna todos os cenários web. Tipos de operadores possíveis: 0 - (padrão) Like; 1 - Equal; 2 - Not like; 3 - Not equal 4 - Exists; 5 - Not exists.
selectHosts	query	Retorne os hosts aos quais o cenário da web pertence como um array na propriedade hosts .
selectSteps	query	Retorne as etapas do cenário da Web na propriedade steps .
selectTags	query	Suporta count. Retorne as tags do cenário web na propriedade tags .
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: httpstestid e name. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no reference commentary .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	

Parâmetro	Tipo	Descrição
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna ou:

- uma série de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando um cenário web

Recupere todos os dados sobre o cenário web "4".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "httptest.get",
  "params": {
    "output": "extend",
    "selectSteps": "extend",
    "httptestids": "9"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "httptestid": "9",
      "name": "Homepage check",
      "nextcheck": "0",
      "delay": "1m",
      "status": "0",
      "variables": [],
      "agent": "Zabbix",
      "authentication": "0",
      "http_user": "",
      "http_password": "",
      "hostid": "10084",
      "templateid": "0",
      "http_proxy": "",
      "retries": "1",
      "ssl_cert_file": "",
      "ssl_key_file": "",
      "ssl_key_password": "",
      "verify_peer": "0",
      "verify_host": "0",
      "headers": [],
      "steps": [
        {
          "httpstepid": "36",
          "httptestid": "9",
          "name": "Homepage",
          "no": "1",

```

```

        "url": "http://example.com",
        "timeout": "15s",
        "posts": "",
        "required": "",
        "status_codes": "200",
        "variables": [
            {
                "name": "{var}",
                "value": "12"
            }
        ],
        "follow_redirects": "1",
        "retrieve_mode": "0",
        "headers": [],
        "query_fields": []
    },
    {
        "httpstepid": "37",
        "httptestid": "9",
        "name": "Homepage / About",
        "no": "2",
        "url": "http://example.com/about",
        "timeout": "15s",
        "posts": "",
        "required": "",
        "status_codes": "200",
        "variables": [],
        "follow_redirects": "1",
        "retrieve_mode": "0",
        "headers": [],
        "query_fields": []
    }
]
},
{
    "id": 1
}
}

```

Veja também

- [Host](#)
- [Scenario step](#)

Fonte

CHttpTest::get() in ui/include/classes/api/services/CHttpTest.php.

httptest.update

Descrição

object httptest.update(object/array webScenarios)

Este método permite atualizar cenários web existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do cenário da Web a serem atualizadas.

A propriedade httptestid deve ser definida para cada cenário web, todas as outras propriedades são opcionais. Somente as propriedades passadas serão

atualizado, todos os outros permanecerão inalterados.

Além das **propriedades padrão do cenário da web**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
steps	array	Cenário etapas para substituir as etapas existentes.
tags	array	Cenário Web tags .

Valores de retorno

(object) Retorna um objeto contendo os IDs dos cenários da web atualizados na propriedade `httptestid`.

Exemplos

Habilitando um cenário web

Habilite um cenário web, ou seja, defina seu status como "0".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "httptest.update",
  "params": {
    "httptestid": "5",
    "status": 0
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "httptestids": [
      "5"
    ]
  },
  "id": 1
}
```

Veja também

- **Scenario step**

Fonte

`CHttpTest::update()` in `ui/include/classes/api/services/CHttpTest.php`.

Configuração

Esta classe é projetada para exportar e importar dados de configuração do Zabbix.

Métodos disponíveis:

- **configuration.export** - exportando a configuração
- **configuration.import** - importando a configuração

configuration.export

Descrição

`string configuration.export(object parameters)`

Este método permite exportar os dados de configuração como uma string serializada.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem os objetos a serem exportados e o formato usar.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **formato**
(obrigatório)|string|Formato no qual os dados devem ser exportados.

Valores possíveis:
yaml - YAML;
xml - XML ;
json - JSON;
raw - array PHP não processado.| |prettyprint|boolean|Torne a saída mais legível adicionando recuo.

Valores possíveis:
true - adiciona recuo;
false - (padrão) não adiciona recuo .| **options**
(obrigatório)|object|Objetos a serem exportados.

O objeto options tem os seguintes parâmetros:
groups - (array) IDs de grupos de hosts para exportar;
hosts - (array) IDs de hosts para exportar;
images - (array) IDs de imagens para exportar;
maps - (array) IDs de mapas a serem exportados;
mediaTypes - (array) IDs de tipos de mídia a serem exportados;
templates - (array) IDs de modelos a serem exportados.
|

Valores de retorno

(string) Retorna uma string serializada contendo os dados de configuração solicitados.

Exemplos**Exportando um host**

Exporta a configuração de um host como uma string XML.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "configuration.export",
  "params": {
    "options": {
      "hosts": [
        "10161"
      ]
    },
    "format": "xml"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n<zabbix_export><version>5.4</version><date>2020",
  "id": 1
}
```

Fonte

CConfiguration::export() em ui/include/classes/api/services/CConfiguration.php.

configuration.import**Descrição**

boolean configuration.import(object parameters)

Este método permite importar dados de configuração de um serializado corda.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros contendo os dados a serem importados e regras de como o dados devem ser tratados.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **|formato**
(obrigatório)|string|Formato da string serializada.

Valores possíveis:
yaml - YAML;
xml - XML;
json - JSON.| **|source**
(obrigatório)|string|String serializada contendo os dados de configuração.| **|rules**
(obrigatório)|object|Regras sobre como objetos novos e existentes devem ser importados.

O parâmetro rules é descrito em detalhes na tabela abaixo.|

Se nenhuma regra for fornecida, a configuração não será Atualizada.

O objeto rules suporta os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| |discoveryRules|object|Regras sobre como importar regras LLD.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novas regras LLD serão criadas; default: false;
updateExisting - (boolean) se definido como true, as regras LLD existentes serão atualizadas; default: false;
deleteMissing - (boolean) se definido como true, as regras LLD não presentes nos dados importados serão excluídas do banco de dados; padrão: falso.| |graphs|object|Regras sobre como importar gráficos.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos gráficos serão criados; default: false;
updateExisting - (boolean) se definido como true, os gráficos existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, os gráficos não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |groups|object|Regras sobre como importar grupos de hosts.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos grupos de hosts serão criados; default: false;
updateExisting - (boolean) se definido como true, os grupos de hosts existentes serão atualizados; padrão: falso.| |hosts|object|Regras sobre como importar hosts.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos hosts serão criados; default: false;
updateExisting - (boolean) se definido como true, os hosts existentes serão atualizados; padrão: falso.| |httpTests|object|Regras sobre como importar cenários da web.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos cenários da web serão criados; default: false;
updateExisting - (boolean) se definido como true, os cenários da web existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, os cenários da web não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |images|object|Regras sobre como importar imagens.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novas imagens serão criadas; default: false;
updateExisting - (boolean) se definido como true, as imagens existentes serão atualizadas; padrão: falso.| |items|object|Regras sobre como importar itens.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos itens serão criados; default: false;
updateExisting - (boolean) se definido como true, os itens existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, os itens não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |maps|object|Regras sobre como importar mapas.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos mapas serão criados; default: false;
updateExisting - (boolean) se definido como true, os mapas existentes serão atualizados; padrão: falso.| |mediaTypes|object|Regras sobre como importar tipos de mídia.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos tipos de mídia serão criados; default: false;
updateExisting - (boolean) se definido como true, os tipos de mídia existentes serão atualizados; padrão: falso.| |templateLinkage|object|Regras sobre como importar links de modelo.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos links entre os modelos e o host serão criados ; default: false;
deleteMissing - (boolean) se definido como true, os links de modelo não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |templates|object|Regras sobre como importar modelos.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos modelos serão criados; default: false;
updateExisting - (boolean) se definido como true, os modelos existentes serão atualizados; padrão: falso.| |templateDashboards|object|Regras sobre como importar painéis de modelo.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos painéis de modelo serão criados; default: false;
updateExisting - (boolean) se definido como true, os painéis de modelos existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, os painéis de modelo não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |triggers|object|Regras sobre como importar gatilhos.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos gatilhos serão criados; default: false;
updateExisting - (boolean) se definido como true, os gatilhos existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, triggers não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.| |valueMaps|object|Regras sobre como importar mapas de valor de host ou modelo.

Parâmetros suportados:
createMissing - (boolean) se definido como true, novos mapas de valor serão criados ; default: false;
updateExisting - (boolean) se definido como true, os mapas de valores existentes serão atualizados; default: false;
deleteMissing - (boolean) se definido como true, os mapas de valores não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.|

Valores de retorno

(boolean) Retorna true se a importação foi bem sucedida.

Exemplos

Importando hosts e itens

Importe o host e os itens contidos na string XML. Se algum item no XML estiver faltando, eles serão deletados do banco de dados e tudo mais será deixado inalterado.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "configuration.import",
  "params": {
    "format": "xml",
    "rules": {
      "valueMaps": {
        "createMissing": true,
        "updateExisting": false
      },
      "hosts": {
        "createMissing": true,
        "updateExisting": true
      },
      "items": {
        "createMissing": true,
        "updateExisting": true,
        "deleteMissing": true
      }
    },
    "source": "<?xml version='1.0' encoding='UTF-8'?>\n<zabbix_export><version>5.4</version><date>"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": true,
  "id": 1
}
```

Fonte

CConfiguration::import() em ui/include/classes/api/services/CConfiguration.php.

configuração.importcompare

Descrição

configuração de matriz.importcompare(parâmetros de objeto)

Este método permite comparar o arquivo de importação com os elementos atuais do sistema e mostra o que será alterado se este arquivo de importação for importado.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros contendo os dados possíveis para importar e regras como os dados devem ser tratados.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **formato**
(obrigatório)|string|Formato da string serializada.

Valores possíveis:
yaml - YAML;
xml - XML;
json - JSON.| **source**
(obrigatório)|string|String serializada contendo os dados de configuração.| **rules**
(obrigatório)|object|Regras sobre como objetos novos e existentes devem ser importados.

O parâmetro rules é descrito em detalhes na tabela abaixo.]

Se nenhuma regra for fornecida, não haverá nada para atualizar e o resultado estará vazio.

A comparação vai ser feita apenas para grupos de hosts e modelos. Triggers e gráficos serão comparados apenas para templates importados, qualquer outro será considerado como "novo".

O objeto rules suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
discoveryRules	object	Regras sobre como importar regras LLD.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novas regras LLD serão criadas; default: false;
updateExisting	(boolean)	se definido como true, as regras LLD existentes serão atualizadas; default: false;
deleteMissing	(boolean)	se definido como true, as regras LLD não presentes nos dados importados serão excluídas do banco de dados; padrão: falso.
graphs	object	Regras sobre como importar gráficos.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos gráficos serão criados; default: false;
updateExisting	(boolean)	se definido como true, os gráficos existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, os gráficos não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
groups	object	Regras sobre como importar grupos de hosts.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos grupos de hosts serão criados; default: false;
updateExisting	(boolean)	se definido como true, os grupos de hosts existentes serão atualizados; padrão: falso.
hosts	object	Regras sobre como importar hosts.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos hosts serão criados; default: false;
updateExisting	(boolean)	se definido como true, os hosts existentes serão atualizados; default: false.
Este parâmetro não fará diferença na saída. É permitido apenas para consistência com configuration.import.		
httpTests	object	Regras sobre como importar cenários da web.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos cenários da web serão criados; default: false;
updateExisting	(boolean)	se definido como true, os cenários da web existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, os cenários da web não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
images	object	Regras sobre como importar imagens.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novas imagens serão criadas; default: false;
updateExisting	(boolean)	se definido como true, as imagens existentes serão atualizadas; default: false.
Este parâmetro não fará diferença na saída. É permitido apenas para consistência com configuration.import.		
items	object	Regras sobre como importar itens.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos itens serão criados; default: false;
updateExisting	(boolean)	se definido como true, os itens existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, os itens não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
maps	object	Regras sobre como importar mapas.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos mapas serão criados; default: false;
updateExisting	(boolean)	se definido como true, os mapas existentes serão atualizados; default: false.
Este parâmetro não fará diferença na saída. É permitido apenas para consistência com configuration.import.		
mediaTypes	object	Regras sobre como importar tipos de mídia.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos tipos de mídia serão criados; default: false;
updateExisting	(boolean)	se definido como true, os tipos de mídia existentes serão atualizados; default: false.
Este parâmetro não fará diferença na saída. É permitido apenas para consistência com configuration.import.		
templateLinkage	object	Regras sobre como importar links de modelo.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos links entre os modelos e o host serão criados; padrão: falso;
deleteMissing	(boolean)	se definido como true, os links de modelo não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
templates	object	Regras sobre como importar modelos.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos modelos serão criados; default: false;
updateExisting	(boolean)	se definido como true, os modelos existentes serão atualizados; padrão: falso.
templateDashboards	object	Regras sobre como importar painéis de modelo.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos painéis de modelo serão criados; default: false;
updateExisting	(boolean)	se definido como true, os painéis de modelos existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, os painéis de modelo não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
triggers	object	Regras sobre como importar gatilhos.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos gatilhos serão criados; default: false;
updateExisting	(boolean)	se definido como true, os gatilhos existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, triggers não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.
valueMaps	object	Regras sobre como importar mapas de valor de host ou modelo.
Parâmetros suportados:		
createMissing	(boolean)	se definido como true, novos mapas de valor serão criados; default: false;
updateExisting	(boolean)	se definido como true, os mapas de valores existentes serão atualizados; default: false;
deleteMissing	(boolean)	se definido como true, os mapas de valores não presentes nos dados importados serão excluídos do banco de dados; padrão: falso.

Valores de retorno

(array) Retorna um array com mudanças na configuração, que serão feitas.

Exemplos

Importando hosts e itens

Importa o template e itens contidos na string YAML. Se qualquer item estiver faltando no YAML, ele será mostrado como excluído e todo o resto permanecerá inalterado.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "configuration.import",
  "params": {
    "format": "xml",
    "rules": {
      "groups": {
        "createMissing": true,
        "updateExisting": true
      },
      "templates": {
        "createMissing": true,
        "updateExisting": true
      },
      "items": {
        "createMissing": true,
        "updateExisting": true,
        "deleteMissing": true
      },
      "triggers": {
        "createMissing": true,
        "updateExisting": true,
        "deleteMissing": true
      },
      "discoveryRules": {
        "createMissing": true,
        "updateExisting": true,
        "deleteMissing": true
      },
      "valueMaps": {
        "createMissing": true,
        "updateExisting": false
      }
    },
    "source": "<?xml version=\"1.0\" encoding=\"UTF-8\"?><zabbix_export><version>5.4</version><date>2020-07-20 12:00:00</date><groups><group name=\"Zabbix servers\"></group></groups><templates><template name=\"Export template\"></template></templates><items><item name=\"\"></item></items></zabbix_export>"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templates": {
      "updated": [
        {
          "before": {
            "uuid": "e1bde9bf2f0544f5929f45b82502e744",
            "template": "Export template",
            "name": "Export template"
          },
          "after": {
            "uuid": "e1bde9bf2f0544f5929f45b82502e744",
            "template": "Export template",
            "name": "Export template"
          }
        },
        {
          "added": [

```

```

{
  "after":{
    "uuid":"3237bc89226e42ed8207574022470e83",
    "name":"Item",
    "key":"item.key",
    "delay":"30s",
    "valuemap":{
      "name":"Host status"
    }
  },
  "triggers":{
    "added":[
      {
        "after":{
          "uuid":"bd1ed0089e4b4f35b762c9d6c599c348",
          "expression":"last(/Export template/item.key)=0",
          "name":"Trigger"
        }
      }
    ]
  }
},
"removed":[
  {
    "before":{
      "uuid":"bd3e7b28b3d544d6a83ed01ddaa65ab6",
      "name":"Old Item",
      "key":"ite_old.key",
      "delay":"30s",
      "valuemap":{
        "name":"Host status"
      }
    }
  }
],
"discovery_rules":{
  "updated":[
    {
      "before":{
        "uuid":"c91616bcf4a44f349539a1b40cb0979d",
        "name":"Discovery rule",
        "key":"rule.key"
      },
      "after":{
        "uuid":"c91616bcf4a44f349539a1b40cb0979d",
        "name":"Discovery rule",
        "key":"rule.key"
      },
      "item_prototypes":{
        "updated":[
          {
            "before":{
              "uuid":"7e164881825744248b3039af3435cf4b",
              "name":"Old item prototype",
              "key":"prototype_old.key"
            },
            "after":{
              "uuid":"7e164881825744248b3039af3435cf4b",
              "name":"Item prototype",
              "key":"prototype.key"
            }
          }
        ]
      }
    }
  ]
}

```

```
}  
}  
}  
}  
]  
}  
]  
}  
}  
]  
}  
}  
]  
},  
"id":1  
}
```

CConfiguration::importcompare() em ui/include/classes/api/services/CConfiguration.php.

Esta classe foi projetada para trabalhar com correlações.

- Correlação

- **correlation.create** - criando novas correlações
- **correlation.delete** - excluindo correlações
- **correlation.get** - recuperando correlações
- **correlation.update** - atualizando correlações

Os objetos a seguir estão diretamente relacionados à API correlation.

O objeto de correlação tem as seguintes propriedades.

Operação de correlação

|Propriedade|Tipo|Descrição| |-----|-----|-----|-----| |tipo
(obrigatório)|inteiro|Tipo de operação.

Valores possíveis:
0 - fechar eventos antigos;
1 - fechar novo evento.

Filtro de correlação

O objeto de filtro de correlação define um conjunto de condições que devem ser met para executar as operações de correlação configuradas. Tem o seguintes propriedades.

Propriedade	Tipo	Descrição
<code>evaltype</code>	<code>
(obrigatório) inteiro</code>	Método de avaliação da condição do filtro. Valores possíveis: <code>
0</code> - e/ou; <code>
1</code> - e; <code>
2</code> - ou; <code>
3</code> - expressão personalizada.
<code>condições</code>	<code>
(obrigatório) array</code>	Conjunto de condições de filtro a serem usadas para filtrar resultados.
<code>eval_formula</code>	<code>string (readonly)</code>	Expressão gerada que será utilizada para avaliar as condições do filtro. A expressão contém IDs que fazem referência a condições de filtro específicas por seu <code>formulaid</code> . O valor de <code>eval_formula</code> é igual ao valor de <code>formula</code> para filtros com uma expressão personalizada.
<code>formula</code>	<code>string</code>	Expressão definida pelo usuário a ser usada para avaliar condições de filtros com uma expressão customizada. A expressão deve conter IDs que façam referência a condições de filtro específicas por seu <code>formulaid</code> . Os IDs usados na expressão devem corresponder exatamente aos definidos nas condições do filtro: nenhuma condição pode permanecer sem uso ou omitida. Obrigatório para filtros de expressão personalizados.

Condição do filtro de correlação

O objeto de condição do filtro de correlação define uma condição específica que deve ser verificado antes de executar as operações de correlação.

Propriedade	**Tipo**	Descrição		-----	-----	-----	-----	**tipo** (obrigatório)	inteiro	Tipo de condição. Valores possíveis: 0 - tag de evento antiga; 1 - nova tag de evento; 2 - novo grupo de host de evento; 3 - par de tags de evento; 4 - valor de tag de evento antigo; 5 - novo valor de tag de evento.
tag	string	tag de evento (antiga ou nova). Necessário quando o tipo de condição é: 0, 1, 4, 5.								
groupid	string	ID do grupo de hosts. Necessário quando o tipo de condição é: 2.								
oldtag	string	Etiqueta de evento antiga. Necessário quando o tipo de condição é: 3.								
newtag	string	Etiqueta de evento antiga. Necessário quando o tipo de condição é: 3.								
value	string	Valor da tag de evento (antiga ou nova). Necessário quando o tipo de condição é: 4, 5.								
formulaid	string	ID exclusivo arbitrário que é usado para fazer referência à condição de uma expressão customizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente.								
operator	inteiro	Operador de condição. Requerido quando o tipo de condição é: 2, 4, 5.								

Para entender melhor como usar filtros com vários tipos de expressões, veja exemplos no [correlation.get](#) e método [correlation.create](#) Páginas.

Os seguintes operadores e valores são suportados para cada condição tipo.

Condição	Nome da condição	Operadores suportados	Valor esperado		-----	-----	-----	-----		2	Grupo de hosts	=, <>	ID do grupo de hosts.
4	Valor da tag de evento antigo	=, <>, like, not like	string										
5	Novo valor da tag de evento	=, <>, like, not like	string										

correlation.delete

Descrição

correlação de objeto.delete(ids de correlação de matriz)

Este método permite excluir correlações.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(array) IDs das correlações a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das correlações sob a propriedade correlationids.

Exemplos

Excluir várias correlações

Exclua duas correlações.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "correlação.excluir",
  "parâmetros": [
    "1",
    "2"
  ],
  "auth": "343baad4f88b4106b9b5961e77437688",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "correlaionídeos": [
```



```

        "1",
        "2"
    ]
},
"id": 1
}

```

Fonte

CCorrelation::delete() em ui/include/classes/api/services/CCorrelation.php.

correlation.get

Descrição

correlação de inteiro/matriz.get(parâmetros de objeto)

O método permite recuperar correlações de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| |correlationids|string/array|Retorna apenas correlações com os IDs fornecidos.| |selectFilter|query|Retorne uma propriedade **filter** com as condições de correlação.| |selectOperations|query|Retorne uma propriedade **operações** com as operações de correlação.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: correlationid, name e status.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos no **comentário de referência**.| |editável|boolean|^| |excludePesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar correlações

Recupere todas as correlações configuradas junto com a correlação condições e operações. O filtro usa o tipo de avaliação "e/ou", então a propriedade formula está vazia e eval_formula é gerado automaticamente.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "correlação.get",
    "parâmetros": {
        "saída": "estender",
        "selectOperations": "extender",
        "selectFilter": "estender"
    },
    "auth": "343baad4f88b4106b9b5961e77437688",
    "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "correlationid": "1",
      "nome": "Correlação 1",
      "Descrição": "",
      "estado": "0",
      "filtro": {
        "evaltype": "0",
        "Fórmula": "",
        "condições": [
          {
            "tipo": "3",
            "oldtag": "erro",
            "newtag": "ok",
            "formulaid": "A"
          }
        ],
        "eval_formula": "A"
      },
      "operações": [
        {
          "tipo": "0"
        }
      ]
    }
  ],
  "id": 1
}

```

Veja também

- [Filtro de correlação](#)
- [Operação de correlação](#)

Fonte

CCorrelation::get() in ui/include/classes/api/services/CCorrelation.php.

correlation.update

Descrição

correlação de objeto.update(correlações de objeto/matriz)

Este método permite atualizar as correlações existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades de correlação a serem atualizadas.

A propriedade correlationid deve ser definida para cada correlação, todas outras propriedades são opcionais. Somente as propriedades passadas serão atualizado, todos os outros permanecerão inalterados.

Além da [correlação padrão properties](#), o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [filter|object|Correlation [filter](#) objeto para substituir o filtro atual.] |operações|matriz|Correlação [operações](#) para substituir as operações existentes.]

Valores de retorno

(object) Retorna um objeto contendo os IDs do correlações sob a propriedade correlationids.

Exemplos

Desativar correlação

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "correlação.atualização",
  "parâmetros": {
    "correlationid": "1",
    "estado": "1"
  },
  "auth": "343baad4f88b4106b9b5961e77437688",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "correlationids": [
      "1"
    ]
  },
  "id": 1
}
```

Substitua as condições, mas mantenha o método de avaliação

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "correlação.atualização",
  "parâmetros": {
    "correlationid": "1",
    "filtro": {
      "condições": [
        {
          "tipo": 3,
          "oldtag": "erro",
          "newtag": "ok"
        }
      ]
    }
  },
  "auth": "343baad4f88b4106b9b5961e77437688",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "correlationids": [
      "1"
    ]
  },
  "id": 1
}
```

Veja também

- [Filtro de correlação](#)
- [Operação de correlação](#)

Fonte

CCorrelation::update() em ui/include/classes/api/services/CCorrelation.php.

#####correlação.criar {#manual-api-reference-correlation-create}

Descrição

correlação de objeto.create(correlações de objeto/matriz)

Este método permite criar novas correlações.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(objeto/array) Correlações a serem criadas.

Além da **correlação padrão properties**, o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| |**operações**
(obrigatório)|array|Correlação **operações** para criar para a correlação.| **filtro**
(obrigatório)|objeto|Correlation **filtro** objeto para a correlação.|

Valores de retorno

(object) Retorna um objeto contendo os IDs do correlações sob a propriedade correlationids. A ordem do os IDs retornados correspondem à ordem das correlações passadas.

Exemplos

Crie uma nova correlação de tag de evento

Crie uma correlação usando o método de avaliação "AND/OR" com uma condição e uma operação. Por padrão, a correlação estará habilitada.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "correlation.create",
  "params": {
    "name": "new event tag correlation",
    "filter": {
      "evaltype": 0,
      "conditions": [
        {
          "type": 1,
          "tag": "ok"
        }
      ]
    },
    "operations": [
      {
        "type": 0
      }
    ]
  },
  "auth": "343baad4f88b4106b9b5961e77437688",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "correlationids": [
      "1"
    ]
  }
}
```

```

    ],
    },
    "id": 1
}

```

Usando um filtro de expressão personalizado

Crie uma correlação que usará uma condição de filtro personalizada. O IDs de fórmula "A" ou "B" foram escolhidos arbitrariamente. O tipo de condição será seja "Grupo de hosts" com o operador "<>".

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "correlação.criar",
  "parâmetros": {
    "name": "nova correlação do grupo de hosts",
    "description": "uma descrição personalizada",
    "estado": 0,
    "filtro": {
      "evaltype": 3,
      "fórmula": "A ou B",
      "condições": [
        {
          "tipo 2",
          "operador": 1,
          "formulaid": "A"
        },
        {
          "tipo 2",
          "operador": 1,
          "formulaid": "B"
        }
      ]
    },
    "operações": [
      {
        "tipo 1"
      }
    ]
  },
  "auth": "343baad4f88b4106b9b5961e77437688",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "correlationids": [
      "2"
    ]
  },
  "id": 1
}

```

Veja também

- [Filtro de correlação](#)
- [Operação de correlação](#)

Fonte

CCorrelation::create() em ui/include/classes/api/services/CCorrelation.php.

Definições

Esta classe foi projetada para funcionar com configurações comuns de administração.

Referências de objetos:

- [Configurações](#)

Métodos disponíveis:

- [settings.get](#) - recupera as configurações
- [settings.update](#) - configurações de atualização

> Objeto de configurações

Os objetos a seguir estão diretamente relacionados à API `settings`.

Definições

O objeto de configurações tem as seguintes propriedades.

Propriedade	Tipo	Descrição
default_lang	string	Idioma do sistema por padrão.
default_timezone	string	Default: <code>en_GB</code> . Fuso horário do sistema por padrão. Padrão: <code>system</code> - padrão do sistema.
default_theme	string	Para obter a lista completa de fusos horários suportados, consulte documentação do PHP . Tema padrão. Valores possíveis: <code>blue-theme</code> - (padrão) Blue; <code>dark-theme</code> - Dark; <code>hc-light</code> - High- luz de contraste; <code>hc-dark</code> - Escuro de alto contraste.
search_limit	integer	Limite para resultados de pesquisa e filtro.
max_overview_table_size	integer	Padrão: 1000. Número máximo de colunas e linhas nos widgets de painel Visão geral de dados e Visão geral do acionador.
max_in_table	integer	Padrão: 50. Contagem máxima de elementos para mostrar dentro da célula da tabela.
server_check_interval	integer	Padrão: 50. Mostrar aviso se o servidor Zabbix estiver inativo.
work_period	string	Valores possíveis: 0 - Não mostrar aviso; 10 - (padrão) Mostrar aviso. Tempo de trabalho. Padrão: 1-5,09:00-18:00.

Propriedade	Tipo	Descrição
show_technical_errors	integer	Mostrar erros técnicos (PHP/SQL) para usuários que não são super admin e para usuários que não fazem parte de grupos de usuários com o modo de depuração ativado. Valores possíveis: 0 - (padrão) Não contém erros técnicos; 1 - Mostra erros técnicos.
history_period	string	Período máximo para exibir dados do histórico nos widgets de painel de dados mais recentes, da Web e de visão geral de dados. Aceita segundos e unidade de tempo com sufixo.
period_default	string	Padrão: 24h. Período padrão do filtro de tempo. Aceita segundos e unidade de tempo com sufixo com suporte para mês e ano (30s,1m,2h,1d,1M,1y).
max_period	string	Padrão: 1h. Período máximo para filtro de tempo. Aceita segundos e unidade de tempo com sufixo com suporte para mês e ano (30s,1m,2h,1d,1M,1y).
severity_color_0	string	Padrão: 2y. Cor da gravidade "Não classificado" como um código de cor hexadecimal.
severity_color_1	string	Padrão: 97AAB3. Cor da gravidade de "Informações" como um código de cor hexadecimal.
severity_color_2	string	Padrão: 7499FF. Cor da gravidade de "Aviso" como um código de cor hexadecimal.
severity_color_3	string	Padrão: FFC859. Cor da gravidade "Média" como um código de cor hexadecimal.
severity_color_4	string	Padrão: FFA059. Cor para gravidade "Alta" como um código de cor hexadecimal.
severity_color_5	string	Padrão: E97659. Cor da gravidade "Desastre" como um código de cor hexadecimal.
severity_name_0	string	Padrão: E45959. Nome da gravidade "Não classificado".
severity_name_1	string	Padrão: Não classificado. Nome para a gravidade de "Informações".
severity_name_2	string	Padrão: informações. Nome para a gravidade de "Aviso".
severity_name_3	string	Padrão: Aviso. Nome para gravidade "Média".
severity_name_4	string	Padrão: Média. Nome para gravidade "Alta".
severity_name_5	string	Padrão: Alta. Nome para a gravidade de "Desastre". Padrão: Desastre.

Propriedade	Tipo	Descrição
custom_color	integer	Use cores personalizadas de status de eventos.
ok_period	string	<p>Valores possíveis: 0 - (padrão) Não use cores personalizadas de status de eventos; 1 - Use cores personalizadas de status de eventos.</p> <p>Exibir período de acionamentos OK. Aceita segundos e unidade de tempo com sufixo.</p>
blink_period	string	<p>Padrão: 5m.</p> <p>Na mudança de status aciona o período de intermitência. Aceita segundos e unidade de tempo com sufixo.</p>
problem_unack_color	string	<p>Padrão: 2m.</p> <p>Cor para eventos PROBLEMA não reconhecidos como um código de cor hexadecimal.</p>
problem_ack_color	string	<p>Padrão: CC0000.</p> <p>Cor para eventos PROBLEMA reconhecidos como um código de cor hexadecimal.</p>
ok_unack_color	string	<p>Padrão: CC0000.</p> <p>Cor para eventos RESOLVIDOS não reconhecidos como um código de cor hexadecimal.</p>
ok_ack_color	string	<p>Padrão: 009900.</p> <p>Cor para eventos RESOLVED reconhecidos como um código de cor hexadecimal.</p>
problem_unack_style	integer	<p>Padrão: 009900.</p> <p>Piscando para eventos de PROBLEMA não confirmados.</p>
problem_ack_style	integer	<p>Valores possíveis: 0 - Não mostra piscando; 1 - (padrão) Mostra piscando.</p> <p>Piscando para eventos de PROBLEMA reconhecidos.</p>
ok_unack_style	integer	<p>Valores possíveis: 0 - Não mostra piscando; 1 - (padrão) Mostra piscando.</p> <p>Piscando para eventos RESOLVIDOS não confirmados.</p>
ok_ack_style	integer	<p>Valores possíveis: 0 - Não mostra piscando; 1 - (padrão) Mostra piscando.</p> <p>Piscando para eventos RESOLVIDOS reconhecidos.</p>
url	string	<p>Valores possíveis: 0 - Não mostra piscando; 1 - (padrão) Mostra piscando.</p> <p>URL de front-end.</p>
discovery_groupid	integer	ID do grupo de hosts ao qual serão colocados automaticamente os hosts descobertos.
default_inventory_mode	integer	Modo de inventário de host padrão.
alert_usrgrpid	integer	<p>Valores possíveis: -1 - (padrão) Desativado; 0 - Manual; 1 - Automático.</p> <p>ID do grupo de usuários para o qual enviará a mensagem de alarme de banco de dados inativo. Se definido como vazio, a mensagem de alarme não será enviada.</p>

Propriedade	Tipo	Descrição
snmptrap_logging	integer	Registrar intercepções SNMP sem correspondência.
login_block	string	<p>Valores possíveis: 0 - Não registrar intercepções SNMP sem correspondência; 1 - (padrão) Registrar intercepções SNMP sem correspondência.</p> <p>Intervalo de tempo durante o qual o formulário de login será bloqueado se o número de tentativas de login com falha exceder o definido no campo login_attempts. Aceita segundos e unidade de tempo com sufixo.</p>
validate_uri_schemes	integer	<p>Padrão: 30s. Validar esquemas de URI.</p>
uri_valid_schemes	string	<p>Valores possíveis: 0 - Não validar; 1 - (padrão) Validar.</p> <p>Esquemas de URI válidos.</p>
x_frame_options	string	<p>Padrão: http,https,ftp,file,mailto,tel,ssh. X-Frame-Options cabeçalho HTTP.</p>
iframe_sandboxing_enabled	integer	<p>Padrão: SAMEORIGIN. Usar iframe sandboxing.</p>
iframe_sandboxing_exceptions	string	<p>Valores possíveis: 0 - Não usar; 1 - (padrão) Usar.</p> <p>Exceções de sandboxing de iframe.</p>
connect_timeout	string	<p>Tempo limite de conexão com o servidor Zabbix.</p>
socket_timeout	string	<p>Padrão: 3s. Tempo limite padrão da rede.</p>
media_type_test_timeout	string	<p>Padrão: 3s. Tempo limite de rede para teste de tipo de mídia.</p>
item_test_timeout	string	<p>Padrão: 65s. Tempo limite de rede para testes de itens.</p>
script_timeout	string	<p>Padrão: 60s. Tempo limite da rede para execução do script.</p>
report_test_timeout	string	<p>Padrão: 60s. Tempo limite da rede para teste de relatório agendado.</p>
auditlog_enabled	integer	<p>Padrão: 60s. Ativar log de auditoria.</p>
ha_failover_delay	string	<p>Valores possíveis: 0 - Desativar; 1 - (padrão) Ativar.</p> <p>Atraso do failover em segundos.</p>
		<p>Padrão: 1m.</p>

Propriedade	Tipo	Descrição
geomaps_tile_provider	string	Provedor de blocos de geomapa. Valores possíveis: OpenStreetMap.Mapnik - (padrão) OpenStreetMap Mapnik; OpenTopoMap - OpenTopoMap; Stamen.TonerLite - Stamen Toner Lite; Stamen.Terrain - Stamen Terrain; USGS.USTopo - USGS US Topo; USGS.USImagery - USGS US Imagery.
geomaps_tile_url	string	Suporta string vazia para especificar valores personalizados de geomaps_tile_url, geomaps_max_zoom e geomaps_attribution. URL do bloco do Geomap se geomaps_tile_provider estiver definido como string vazia.
geomaps_max_zoom	integer	Nível de zoom máximo do Geomap se geomaps_tile_provider estiver definido como string vazia. O zoom máximo deve estar no intervalo entre 0 e 30.
geomaps_attribution	string	Texto de atribuição do Geomap se geomaps_tile_provider estiver definido como string vazia.

settings.get

Descrição

configurações do objeto.get(parâmetros do objeto)

O método permite recuperar o objeto de configurações de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta apenas um parâmetro.

Parâmetro	Tipo	Descrição
output	query	Este parâmetro é comum para todos os métodos get descritos no comentário de referência .

Valores de retorno

(object) Retorna o objeto de configurações.

Exemplos

Request:

```
{
  "jsonrpc": "2.0",
  "method": "settings.get",
  "params": {
    "output": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "default_theme": "blue-theme",
    "search_limit": "1000",
    "max_in_table": "50",
    "server_check_interval": "10",
    "work_period": "1-5,09:00-18:00",
    "show_technical_errors": "0",
    "history_period": "24h",
    "period_default": "1h",
    "max_period": "2y",
    "severity_color_0": "97AAB3",
    "severity_color_1": "7499FF",
    "severity_color_2": "FFC859",
    "severity_color_3": "FFA059",
    "severity_color_4": "E97659",
    "severity_color_5": "E45959",
    "severity_name_0": "Not classified",
    "severity_name_1": "Information",
    "severity_name_2": "Warning",
    "severity_name_3": "Average",
    "severity_name_4": "High",
    "severity_name_5": "Disaster",
    "custom_color": "0",
    "ok_period": "5m",
    "blink_period": "2m",
    "problem_unack_color": "CC0000",
    "problem_ack_color": "CC0000",
    "ok_unack_color": "009900",
    "ok_ack_color": "009900",
    "problem_unack_style": "1",
    "problem_ack_style": "1",
    "ok_unack_style": "1",
    "ok_ack_style": "1",
    "discovery_groupid": "5",
    "default_inventory_mode": "-1",
    "alert_usrgrpid": "7",
    "snmptrap_logging": "1",
    "default_lang": "en_GB",
    "default_timezone": "system",
    "login_attempts": "5",
    "login_block": "30s",
    "validate_uri_schemes": "1",
    "uri_valid_schemes": "http,https,ftp,file,mailto,tel,ssh",
    "x_frame_options": "SAMEORIGIN",
    "iframe_sandboxing_enabled": "1",
    "iframe_sandboxing_exceptions": "",
    "max_overview_table_size": "50",
    "connect_timeout": "3s",
    "socket_timeout": "3s",
    "media_type_test_timeout": "65s",
    "script_timeout": "60s",
    "item_test_timeout": "60s",
    "url": "",
    "report_test_timeout": "60s",
    "auditlog_enabled": "1",
    "ha_failover_delay": "1m",
    "geomaps_tile_provider": "OpenStreetMap.Mapnik",
    "geomaps_tile_url": "",
    "geomaps_max_zoom": "0",
    "geomaps_attribution": ""
  }
}

```

```
    },  
    "id": 1  
}
```

Fonte

CSettings::get() in ui/include/classes/api/services/CSettings.php.

settings.update

Descrição

configurações do objeto.update(configurações do objeto)

Este método permite atualizar as configurações comuns existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Propriedades de configurações a serem atualizadas.

Valores de retorno

(array) Retorna array com os nomes dos parâmetros atualizados.

Exemplos

Request:

```
{  
  "jsonrpc": "2.0",  
  "method": "settings.update",  
  "params": {  
    "login_attempts": "1",  
    "login_block": "1m"  
  },  
  "auth": "038e1d7b1735c6a5436ee9eae095879e",  
  "id": 1  
}
```

Response:

```
{  
  "jsonrpc": "2.0",  
  "result": [  
    "login_attempts",  
    "login_block"  
  ],  
  "id": 1  
}
```

Fonte

CSettings::update() in ui/include/classes/api/services/CSettings.php.

Evento

Esta classe foi projetada para trabalhar com eventos.

Referências de objetos:

- [Evento](#)

Métodos disponíveis:

- [event.get](#) - recuperando eventos
- [evento.reconhecimento](#) - reconhecendo eventos

> Objeto de evento

Os objetos a seguir estão diretamente relacionados à API event.

Evento

Note:

Os eventos são criados pelo servidor Zabbix e não podem ser modificados por meio da API.

O objeto de evento tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição| |-----|-----|-----| |eventid|string|ID do evento. | |source|integer|Tipo do evento.

Valores possíveis:
0 - evento criado por um acionador;
1 - evento criado por uma regra de descoberta;
2 - evento criado por registro automático de agente ativo;
3 - evento interno;
4 - evento criado na atualização do status do serviço. | |objeto|integer|Tipo de objeto relacionado ao evento.

Valores possíveis para eventos de acionamento:
0 - acionador.

Valores possíveis para eventos de descoberta:
1 - host descoberto;
2 - serviço descoberto.

Valores possíveis para eventos de registro automático:
3 - host registrado automaticamente.

Valores possíveis para eventos internos:
0 - gatilho;
4 - item;
5 - regra LLD.

Valores possíveis para eventos de serviço:
6 - serviço. | |objectid|string|ID do objeto relacionado. | |acknowledged|integer|Se o evento foi reconhecido. | |clock|timestamp|Horário em que o evento foi criado. | |ns|integer|Nanosegundos quando o evento foi criado. | |name|string|Nome do evento resolvido. | |valor|integer|Estado do objeto relacionado.

Valores possíveis para eventos de acionamento e serviço:
0 - OK;
1 - problema.

Valores possíveis para eventos de descoberta:
0 - host ou serviço ativo;
1 - host ou serviço desativado;
2 - host ou serviço descoberto;
3 - host ou serviço perdido.

Valores possíveis para eventos internos:
0 - estado "normal";
1 - estado "desconhecido" ou "não suportado".

Este parâmetro não é usado para eventos de registro automático de agente ativo. | |severity|integer|Gravidade atual do evento.

Valores possíveis:
0 - não classificado;
1 - informação;
2 - aviso;
3 - média;
4 - alto;
5 - desastre. | |r_eventid|string|ID do evento de recuperação | |c_eventid|string|ID do evento que foi usado para substituir (fechar) o evento atual sob a regra de correlação global. Consulte [correlationid](#) para identificar a regra de correlação exata.

Este parâmetro só é definido quando o evento é fechado pela regra de correlação global. | |correlationid|string|ID da regra de correlação que gerou o fechamento do problema.

Este parâmetro só é definido quando o evento é fechado pela regra de correlação global. | |userid|string|ID do usuário se o evento foi fechado manualmente. | |suppressed|integer|Se o evento é suprimido.

Valores possíveis:
0 - o evento está em estado normal;
1 - o evento está suprimido. | |opdata|string|Dados operacionais com macros expandidas. | |urls|matriz de [URLs de tipo de mídia](#)|URLs de tipos de mídia ativos.]

Tag de evento

O objeto de tag de evento tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição| |-----|-----|-----| |tag|string|Nome da tag do evento. | |valor|string|Valor da tag do evento.]

URLs de tipo de mídia

Objeto com URL de tipo de mídia tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição| |-----|-----|-----| |name|string|Nome do URL definido pelo tipo de mídia. | |url|string|Valor de URL definido pelo tipo de mídia.]

Os resultados conterão entradas apenas para tipos de mídia ativos com entrada do menu de eventos. A macro usada nas propriedades será expandida, mas se uma das propriedades contém macro não expandida ambas as propriedades serão excluídas dos resultados. Macros compatíveis descritas em [página](#).

event.acknowledge

Descrição

object event.acknowledge (parâmetros de objeto/array)

Este método permite atualizar eventos. As seguintes ações de atualização podem ser realizadas:

- Fechar evento. Se o evento já foi resolvido, esta ação será ignorada.

- Reconhecer evento. Se o evento já foi reconhecido, esta ação será ignorado.
- Adicionar mensagem.
- Alterar a gravidade do evento. Se o evento já tiver a mesma gravidade, esta ação será ignorada.

::: não importante Somente eventos de gatilho podem ser atualizados.

Somente eventos de problema podem ser atualizados.

Os direitos de leitura/gravação para o gatilho são necessários para fechar o evento ou para alterar a gravidade do evento.

Para fechar um evento, o fechamento manual deve ser permitido no acionar. :::

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Parâmetros contendo os IDs dos eventos e atualização operações que devem ser realizadas.

[Parâmetro|Tipo|Descrição| |-----|-----|-----|-----| |eventids
(obrigatório)|string/object|IDs dos eventos a serem reconhecidos.| |ação
(obrigatório)|inteiro|Ação(ões) de atualização do evento. Este é um campo de máscara de bits, qualquer combinação de valores é aceitável.

Valores possíveis:
1 - fechar problema;
2 - reconhecer evento;
4 - adicionar mensagem;
8 - alterar a gravidade;
16 - evento de não confirmação.| |message|string|Texto da mensagem.

Requerido, se a ação contiver o sinalizador 'adicionar mensagem'.| |severity|integer|Nova gravidade para eventos.

Obrigatório, se a ação contiver o sinalizador 'alterar gravidade'.

Valores possíveis:
0 - não classificado;
1 - informações;
2 - aviso;
3 - média;
4 - alta;
5 - desastre.]

Valores de retorno

(object) Retorna um objeto contendo os IDs dos eventos atualizados sob a propriedade eventids.

Exemplos

Reconhecendo um evento

Reconheça um único evento e deixe uma mensagem.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "evento.reconhecimento",
  "parâmetros": {
    "eventos": "20427",
    "ação": 6,
    "message": "Problema resolvido."
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "eventos": [
      "20427"
    ]
  },
  "id": 1
}
```

Alterando a gravidade do evento

Altere a gravidade de vários eventos e deixe uma mensagem.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "evento.reconhecimento",
```

```

    "parâmetros": {
        "eventos": ["20427", "20428"],
        "ação": 12,
        "message": "Manutenção necessária para corrigi-lo.",
        "gravidade": 4
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "eventos": [
            "20427",
            "20428"
        ]
    },
    "id": 1
}

```

Fonte

CEvent::acknowledge() em ui/include/classes/api/services/CEvent.php.

event.get

Descrição

integer/array event.get(object parameters)

O método permite recuperar eventos de acordo com os parâmetros fornecidos.

::: não importante Este método pode retornar eventos de uma entidade excluída se esses eventos não foram removidos pela governanta ainda. :::

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| |eventids|string/array|Retorna apenas eventos com os IDs fornecidos.| |groupids|string/array|Retorna apenas eventos criados por objetos que pertencem aos grupos de hosts especificados.| |hostids|string/array|Retorna apenas eventos criados por objetos que pertencem aos hosts fornecidos.| |objectids|string/array|Retorna apenas eventos criados pelos objetos fornecidos.| |source|integer|Retorna apenas eventos com o tipo fornecido.

Consulte a [página do objeto do evento](#) para obter uma lista de tipos de eventos suportados.

Padrão: 0 - acionar eventos.| |objeto|integer|Retorna apenas eventos criados por objetos de um determinado tipo.

Consulte a [página do objeto do evento](#) para obter uma lista de tipos de objetos compatíveis.

Padrão : 0 - gatilho.| |acknowledged|boolean|Se definido como true retorna apenas eventos reconhecidos.| |suppressed|boolean|true - retorna somente eventos suprimidos;
false - retorna eventos no estado normal.| |severities|integer/array|Retorna apenas eventos com as severidades de eventos especificadas. Aplica-se apenas se o objeto for acionador.| |evaltype|integer|Regras para pesquisa de tags.

Valores possíveis:
0 - (padrão) E/Ou;
2 - Ou.| |tags|array de objetos|Retorna apenas eventos com as tags fornecidas. Correspondência exata por tag e pesquisa sem distinção entre maiúsculas e minúsculas por valor e operador.
Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator> "}, ...].
Um array vazio retorna todos os eventos.

Possíveis tipos de operadores:
0 - (padrão) Like;
1 - Igual;
2 - Não gosta;
3 - Não é igual
4 - Existe;
5 - Não existe.| |eventid_from|string|Retorna apenas eventos com IDs maiores ou iguais ao ID fornecido.| |eventid_till|string|Retorna apenas eventos com IDs menores ou iguais ao ID fornecido.| |time_from|timestamp|Retorna apenas eventos que foram criados após ou no momento determinado.| |time_till|timestamp|Retorna

apenas eventos que foram criados antes ou no momento determinado. | `|problem_time_from|timestamp|` Retorna apenas eventos que estavam no estado de problema começando com `problem_time_from`. Aplica-se apenas se a origem for um evento de acionamento e o objeto for um acionador. Obrigatório se `problem_time_till` for especificado. | `|problem_time_till|timestamp|` Retorna apenas eventos que estavam no estado de problema até `problem_time_till`. Aplica-se apenas se a origem for um evento de acionamento e o objeto for um acionador. Obrigatório se `problem_time_from` for especificado. | `|valor|inteiro/array|` Retorna apenas eventos com os valores fornecidos. | `|selectHosts|query|` Retorne uma propriedade `hosts` com hosts contendo o objeto que criou o evento. Com suporte apenas para eventos gerados por gatilhos, itens ou regras LLD. | `|selectRelatedObject|query|` Retorna uma propriedade `relatedObject` com o objeto que criou o evento. O tipo de objeto retornado depende do tipo de evento. | `|select_alerts|query|` Retorna uma propriedade `alerts` com alertas gerados pelo evento. Os alertas são classificados em ordem cronológica inversa. | `|select_acknowledges|query|` Retorna uma propriedade `acknowledges` com atualizações de eventos. As atualizações de eventos são classificados em ordem cronológica inversa.

O objeto de atualização de evento tem as seguintes propriedades:

`acknowledgeid` - ID da confirmação de (string);

`userid` - (string) ID do usuário que atualizou o evento;

`eventid` - (string) ID do evento atualizado;

`clock` - (timestamp) hora em que o evento foi atualizado;

`message` - texto (string) da mensagem;

`action` - (integer) ação de atualização que foi executada veja `event.acknowledge`;

`old_severity` - gravidade do evento (integer) antes desta ação de atualização;

`new_severity` - gravidade do evento (integer) após esta ação de atualização;

`username` - (string) nome de usuário do usuário que atualizou o evento;

`name` - (string) nome do usuário que atualizou o evento;

`surname` - (string) sobrenome do usuário que atualizou o evento.

Suporta contagem. | `|selectTags|query|` Retorne uma propriedade `tags` com tags de evento. | `|selectSuppressionData|query|` Retorna uma propriedade `suppression_data` com a lista de manutenções:

`maintenanceid` - (string) ID da manutenção;

`suppress_until` - (integer) tempo até o evento é suprimido. | `|sortfield|string/array|` Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: `eventid`, `objectid` e `clock`. | `|countOutput|boolean|` Esses parâmetros sendo comuns para todos os métodos `get` são descritos em detalhes na página [comentário de referência](#). | `|editável|booleano|^|` | `|excluirPesquisa|boolean|^|` | `|filtro|objeto|^|` | `|limite|inteiro|^|` | `|saída|consulta|^|` | `|preservekeys|boolean|^|` | `|pesquisar|objeto|^|` | `|searchByAny|boolean|^|` | `|searchWildcardsEnabled|boolean|^|` | `|ordem de classificação|cadeia/matriz|^|` | `|startSearch|boolean|^|`

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro `countOutput` tiver sido usado.

Exemplos

Recuperando eventos de gatilho

Recupere os eventos mais recentes do gatilho "13926".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "evento.get",
  "parâmetros": {
    "saída": "estender",
    "select_acknowledges": "estender",
    "selectTags": "estender",
    "selectSuppressionData": "estender",
    "objectids": "13926",
    "sortfield": ["relógio", "id do evento"],
    "ordem de classificação": "DESC"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "eventid": "9695",
      "fonte": "0",
      "objeto": "0",
      "objeto": "13926",
      "relógio": "1347970410",
```



```

    "valor": "1",
    "reconhecido": "1",
    "ns": "413316245",
    "name": "MySQL está inativo",
    "gravidade": "5",
    "r_eventid": "0",
    "c_eventid": "0",
    "correlationid": "0",
    "userid": "0",
    "opdata": "",
    "reconhece": [
      {
        "reconhecimento": "1",
        "userid": "1",
        "eventid": "9695",
        "relógio": "1350640590",
        "message": "Problema resolvido.\n\r----[BULK ACKNOWLEDGE]----",
        "ação": "6",
        "old_severity": "0",
        "new_severity": "0",
        "username": "Admin",
        "nome": "Zabbix",
        "sobrenome": "Administrador"
      }
    ],
    "suppression_data": [
      {
        "manutenção": "15",
        "suppress_until": "1472511600"
      }
    ],
    "suprimido": "1",
    "Tag": [
      {
        "tag": "serviço",
        "valor": "mysqld"
      },
      {
        "tag": "erro",
        "valor": ""
      }
    ]
  },
  {
    "evento": "9671",
    "fonte": "0",
    "objeto": "0",
    "objeto": "13926",
    "relógio": "1347970347",
    "valor": "0",
    "reconhecido": "0",
    "ns": "0",
    "name": "Indisponível por ping ICMP",
    "gravidade": "4",
    "r_eventid": "0",
    "c_eventid": "0",
    "correlationid": "0",
    "userid": "0",
    "opdata": "",
    "reconhece": [],
    "suppression_data": [],
    "suprimido": "0",

```

```

        "Tag": []
    }
],
    "id": 1
}

```

Recuperando eventos por período de tempo

Recupere todos os eventos que foram criados entre 9 e 10 de outubro, 2012, em ordem cronológica inversa.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "evento.get",
    "parâmetros": {
        "saída": "estender",
        "time_from": "1349797228",
        "time_till": "1350661228",
        "sortfield": ["relógio", "id do evento"],
        "sortorder": "desc"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": [
        {
            "eventid": "20616",
            "fonte": "0",
            "objeto": "0",
            "objeto": "14282",
            "relógio": "1350477814",
            "valor": "1",
            "reconhecido": "0",
            "ns": "0",
            "name": "Menos de 25% livre no cache do histórico",
            "gravidade": "3",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "opdata": "",
            "suprimido": "0"
        },
        {
            "eventid": "20617",
            "fonte": "0",
            "objeto": "0",
            "objeto": "14283",
            "relógio": "1350477814",
            "valor": "0",
            "reconhecido": "0",
            "ns": "0",
            "name": "0 Zabbix Trapper processa mais de 75% ocupado",
            "gravidade": "3",
            "r_eventid": "0",
            "c_eventid": "0",
            "correlationid": "0",
            "userid": "0",
            "opdata": ""
        }
    ]
}

```

```

        "suprimido": "0"
    },
    {
        "eventid": "20618",
        "fonte": "0",
        "objeto": "0",
        "objeto": "14284",
        "relógio": "1350477815",
        "valor": "1",
        "reconhecido": "0",
        "ns": "0",
        "name": "Alta perda de ping ICMP",
        "gravidade": "3",
        "r_eventid": "0",
        "c_eventid": "0",
        "correlationid": "0",
        "userid": "0",
        "opdata": "",
        "suprimido": "0"
    }
],
"id": 1
}

```

Veja também

- [Alerta](#)
- [Item](#)
- [Host](#)
- [Regra LLD](#)
- [Trigger](#)

Fonte

CEvent::get() em ui/include/classes/api/services/CEvent.php.

Expressão regular

Esta classe foi projetada para trabalhar com expressões regulares globais.

Referências de objetos:

- [Expressão regular](#)

Métodos disponíveis:

- [regexp.create](#) - criando novas expressões regulares
- [regexp.delete](#) - excluindo expressões regulares
- [regexp.get](#) - recuperando expressões regulares
- [regexp.update](#) - atualizando expressões regulares

> Objeto de expressão regular

Os objetos a seguir estão diretamente relacionados à API `regexp`.

Expressão regular

O objeto de expressão regular global tem as seguintes propriedades.

Propriedade	Tipo	Descrição
<code>regexpid</code>	string	(somente leitura) ID da expressão regular.
<code>name</code> (requerido)	string	Nome da expressão regular.

Propriedade	Tipo	Descrição
test_string	string	Sequência de teste.

Objeto de expressões

O objeto de expressões tem as seguintes propriedades.

Propriedade	Tipo	Descrição
expression (required)	string	Expressão regular.
expression_type (required)	integer	Tipo de expressão regular. Valores possíveis: 0 - Cadeia de caracteres incluída; 1 - Qualquer cadeia de caracteres incluída; 2 - Cadeia de caracteres não incluída; 3 - O resultado é TRUE; 4 - O resultado é FALSE.
exp_delimiter	string	Delimitador de expressão. Somente quando expression_type Qualquer string de caracteres incluída. Valor padrão ,.
case_sensitive	integer	Valores possíveis: , , , /. Diferenciação de maiúsculas e minúsculas. Valor padrão "0". Valores possíveis: 0 - Não diferencia maiúsculas de minúsculas; 1 - Diferencia maiúsculas de minúsculas.

regexp.create

Descrição

`object regexp.create(object/array regularExpressions)`

Este método permite criar novas expressões regulares globais.

Note:

Este método está disponível apenas para os tipos de usuário Super admin.
As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Veja [Usuário roles](#) para obter mais informações.

Parâmetros

(object/array) Expressões regulares para criar.

Além das **propriedades padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
expressions	array	Expressões opções

Valores de retorno

(object) Retorna um objeto contendo os IDs das expressões regulares criadas sob a propriedade `regexpids`.

Exemplos

Criando uma nova expressão regular global.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "regex.create",
  "params": {
    "name": "Storage devices for SNMP discovery",
    "test_string": "/boot",
    "expressions": [
      {
        "expression": "^(Physical memory|Virtual memory|Memory buffers|Cached memory|Swap space)$",
        "expression_type": "4",
        "case_sensitive": "1"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "regexpids": [
      "16"
    ]
  },
  "id": 1
}
```

Fonte

CRegex::create() in ui/include/classes/api/services/CRegex.php.

regex.delete

Descrição

object regex.delete(array regexpids)

Este método permite deletar expressões regulares globais.

Note:

Este método está disponível apenas para os tipos de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs das expressões regulares a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das expressões regulares excluídas sob a propriedade regexpids

Exemplos

Excluindo várias expressões regulares globais.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "regex.delete",
  "params": [
    "16",
    "17"
  ],
}
```

```

    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "regexids": [
      "16",
      "17"
    ]
  },
  "id": 1
}

```

Fonte

CRegex::delete() in ui/include/classes/api/services/CRegex.php.

regex.get

Descrição

integer/array regex.get(object parameters)

O método permite recuperar expressões regulares globais de acordo com os parâmetros fornecidos.

Note:

Este método está disponível apenas para Super Admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
regexids	string/array	Retorna apenas expressões regulares com os IDs fornecidos.
selectExpressions	query	Retorna uma propriedade expressions .
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: regexpid e name. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;

- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando expressões regulares globais.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "regexp.get",
  "params": {
    "output": ["regexpid", "name"],
    "selectExpressions": ["expression", "expression_type"],
    "regexpids": [1, 2],
    "preservekeys": true
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "1": {
      "regexpid": "1",
      "name": "File systems for discovery",
      "expressions": [
        {
          "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|ntfs|fat32)",
          "expression_type": "3"
        }
      ]
    },
    "2": {
      "regexpid": "2",
      "name": "Network interfaces for discovery",
      "expressions": [
        {
          "expression": "^Software Loopback Interface",
          "expression_type": "4"
        },
        {
          "expression": "^(In)?[Ll]oop[Bb]ack[0-9._]*$",
          "expression_type": "4"
        },
        {
          "expression": "^NULL[0-9.]*$",
          "expression_type": "4"
        },
        {
          "expression": "^[Ll]o[0-9.]*$",
          "expression_type": "4"
        },
        {
          "expression": "^[Ss]ystem$",
          "expression_type": "4"
        },
        {
          "expression": "^Nu[0-9.]*$",
          "expression_type": "4"
        }
      ]
    }
  }
}
```

```

    },
    "id": 1
}

```

Fonte

CRegexp::get() in ui/include/classes/api/services/CRegexp.php.

regex.update

Descrição

object regex.update(object/array regularExpressions)

Este método permite atualizar as expressões regulares globais existentes.

Note:

Este método está disponível apenas para os tipos de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades de expressão regular a serem atualizadas.

A propriedade regexpid deve ser definida para cada objeto, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das **propriedades padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
expressions	array	Expressões opções.

Valores de retorno

(object) Retorna um objeto contendo os IDs das expressões regulares atualizadas na propriedade regexpids

Exemplos

Atualização da expressão regular global para descoberta de sistemas de arquivos.

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "regex.update",
  "params": {
    "regexpid": "1",
    "name": "File systems for discovery",
    "test_string": "",
    "expressions": [
      {
        "expression": "^(btrfs|ext2|ext3|ext4|reiser|xfs|ffs|ufs|jfs|jfs2|vxfs|hfs|apfs|refs|zfs)$",
        "expression_type": "3",
        "exp_delimiter": ",",
        "case_sensitive": "0"
      },
      {
        "expression": "^(ntfs|fat32|fat16)$",
        "expression_type": "3",
        "exp_delimiter": ",",
        "case_sensitive": "0"
      }
    ]
  },
  "auth": "700ca65537074ec963db7efabda78259",

```



```
}
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "regexpids": [
      "1"
    ]
  },
  "id": 1
}
```

Fonte

CRegexp::update() in ui/include/classes/api/services/CRegexp.php.

Função

Esta classe foi projetada para trabalhar com funções de usuário.

Referências de objetos:

- [Função](#)
- [Regras de função](#)
- [Elemento de IU](#)
- [Serviço](#)
- [Etiqueta de serviço](#)
- [Módulo](#)
- [Ação](#)

Métodos disponíveis:

- [role.create](#) - criar novas funções de usuário
- [role.delete](#) - excluir funções de usuário
- [role.get](#) - recuperar funções de usuário
- [role.update](#) - atualizar funções de usuário

> Objeto de função

Os objetos a seguir estão diretamente relacionados à API `role`.

Função

O objeto `role` tem as seguintes propriedades:

Propriedade	Tipo	Descrição
<code>roleid</code>	string	(somente leitura) ID da função.
<code>name</code> (requerido)	string	Nome da role.
<code>type</code> (requerido)	integer	Tipo de usuário. Valores possíveis: 1 - (padrão) Usuário; 2 - Admin; 3 - Super admin.
<code>readonly</code>	integer	(somente leitura) Se a função é somente leitura. Valores possíveis: 0 - (padrão) Não; 1 - Sim.

Regras de função

O objeto de regras de função tem as seguintes propriedades:

Propriedade	Tipo	Descrição
ui	array	Array dos objetos [elemento da interface do usuário](object#ui_element.
ui.default_access	integer	Se o acesso a novos elementos da interface do usuário está ativado.
services.read.mode	integer	<p>Valores possíveis:</p> <p>0 - Desativado;</p> <p>1 - (padrão) Ativado.</p> <p>Acesso somente leitura aos serviços.</p> <p>Valores possíveis:</p> <p>0 - Acesso somente leitura aos serviços, especificado pelo <code>services.read.list</code> ou correspondido pelo <code>services.read</code> Propriedades <code>.tag</code>.</p> <p>1 - (padrão) Acesso somente leitura a todos os serviços.</p>
services.read.list	array	<p>Array de objetos Serviço.</p> <p>Os serviços especificados, incluindo serviços filho, terão acesso somente leitura à função de usuário. O acesso somente leitura não substituirá o acesso de leitura e gravação aos serviços.</p>
services.read.tag	object	<p>Usado somente se <code>services.read.mode</code> estiver definido como 0.</p> <p>Array do objeto Tag de serviço.</p> <p>Os serviços correspondentes à tag, incluindo serviços filho, terão acesso somente leitura à função de usuário. O acesso somente leitura não substituirá o acesso de leitura e gravação aos serviços.</p>
services.write.mode	integer	<p>Usado somente se <code>services.read.mode</code> for definido como 0.</p> <p>Acesso de leitura e gravação aos serviços.</p> <p>Valores possíveis:</p> <p>0 - (padrão) Acesso de leitura e gravação aos serviços, especificados pelo <code>services.write.list</code> ou correspondidos pelo <code>services.write.tag</code>.</p> <p>1 - Acesso de leitura e gravação a todos os serviços.</p>
services.write.list	array	<p>Array de objetos Serviço.</p> <p>Os serviços especificados, incluindo serviços filho, terão acesso de leitura e gravação à função de usuário. O acesso de leitura e gravação substituirá o acesso somente leitura aos serviços.</p>
services.write.tag	object	<p>Usado somente se <code>services.write.mode</code> for definido como 0.</p> <p>Array do objeto Tag de serviço.</p> <p>Os serviços correspondentes à tag, incluindo serviços filho, terão acesso de leitura e gravação à função de usuário. O acesso de leitura e gravação substituirá o acesso somente leitura aos serviços.</p>
modules	array	Usado somente se <code>services.write.mode</code> for definido como 0.
modules.default_access	integer	<p>Array dos objetos módulo.</p> <p>Se o acesso a novos módulos está ativado.</p> <p>Valores possíveis:</p> <p>0 - Desativado;</p> <p>1 - (padrão) Ativado.</p>

Propriedade	Tipo	Descrição
api.access	integer	Se o acesso à API está ativado. Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.
api.mode	integer	Modo de tratamento de métodos de API listados na propriedade api. Valores possíveis: 0 - (padrão) Lista de negações; 1 - Lista de permissões.
api	array	Array de métodos de API.
actions	array	Array dos objetos ação .
actions.default_access	integer	Se o acesso a novas ações está ativado. Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.

Elemento de IU

O objeto de elemento de interface do usuário (UI) tem as seguintes propriedades:

Propriedade	Tipo	Descrição
name (requerido)	string	<p>Nome do elemento da interface do usuário.</p> <p>Valores possíveis para usuários de qualquer tipo:</p> <p>monitoring.dashboard - Monitoramento → Dashboard; monitoring.problems - Monitoramento → Problemas; monitoring.hosts - Monitoramento → Hosts; monitoring.overview - Monitoramento → Visão geral; monitoring.latest_data - Monitoramento → Dados recentes; monitoring.maps - Monitoramento → Mapas; monitoring.services - Monitoramento → Serviços; inventory.overview - Inventário → Visão geral; inventory.hosts - Inventário → Hosts; reports.availability_report - Relatórios → Relatório de disponibilidade; reports.top_triggers - Relatórios → Triggers top 100.</p> <p>Valores possíveis apenas para usuários dos tipos de usuário Admin e Super admin:</p> <p>monitoring.discovery - Monitoramento → Descoberta; reports.scheduled_reports - Relatórios → Relatórios programados; reports.notifications - Relatórios → Notificações; configuration.host_groups - Configuração → Grupos de hosts; configuration.templates - Configuração → Templates; configuration.hosts - Configuração → Hosts; configuration.maintenance - Configuração → Manutenção; configuration.actions - Configuração → Ações; configuration.discovery - Configuração → Descoberta.</p> <p>Valores possíveis apenas para usuários do tipo de usuário Super admin:</p> <p>reports.system_info - Relatórios → Informação do sistema; reports.audit - Relatórios → Auditoria; reports.action_log - Relatórios → Log de ações; configuration.event_correlation - Configuration → Correlação de eventos; administration.general - Administração → Geral; administration.proxies - Administração → Proxies; administration.authentication - Administração → Autenticação; administration.user_groups - Administração → User groups; administration.user_roles - Administração → Funções do usuário; administration.users - Administração → Usuários; administration.media_types - Administração → Tipos de mídia; administration.scripts - Administração → Scripts; administration.queue - Administração → Fila.</p>
status	integer	<p>Se o acesso ao elemento da interface do usuário está ativado.</p> <p>Valores possíveis:</p> <p>0 - Desativado; 1 - (padrão) Ativado.</p>

Serviço

Propriedade	Tipo	Descrição
serviceid (requerido)	string	ID do serviço.

Etiqueta de serviço

Propriedade	Tipo	Descrição
tag (requerido)	string	Nome da etiqueta. Se uma string vazia for especificada, a etiqueta de serviço não será usada para correspondência de serviço.
value	string	Valor da etiqueta. Se nenhum valor ou string vazia for especificado, apenas o nome da etiqueta será usado para correspondência de serviço.

Módulo

O objeto módulo tem as seguintes propriedades:

Propriedade	Tipo	Descrição
moduleid (requerido)	string	ID do módulo.
status	integer	Se o acesso ao módulo está ativado. Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.

Ação

O objeto de ação tem as seguintes propriedades:

Propriedade	Tipo	Descrição
name (requerido)	string	Nome da ação. Valores possíveis para usuários de qualquer tipo: edit_dashboards - Criar e editar dashboards; edit_maps - Criar e editar mapas; add_problem_comments - Adicionar comentários do problema; change_severity - Alterar a gravidade do problema; acknowledge_problems - Reconhecer problemas; close_problems - Fechar problemas; execute_scripts - Executar scripts; manage_api_tokens - Gerenciar tokens de API.
status	integer	Valores possíveis apenas para usuários dos tipos de usuário Admin e Super admin: edit_maintenance - Criar e editar manutenções; manage_scheduled_reports - Gerenciar relatórios programados. Se o acesso para realizar a ação está ativado. Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.

role.create

Descrição

`object role.create(object/array roles)`

Este método permite criar novas funções.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Funções para criar.

Além das [propriedades de função padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
rules	array	Função rules a ser criada para a função.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos papéis criados na propriedade roleids. A ordem dos IDs retornados corresponde à ordem das funções passadas.

Exemplos**Criando uma função**

Crie uma função com o tipo "Usuário" e acesso negado a dois elementos da interface do usuário.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "role.create",
  "params": {
    "name": "Operator",
    "type": "1",
    "rules": {
      "ui": [
        {
          "name": "monitoring.hosts",
          "status": "0"
        },
        {
          "name": "monitoring.maps",
          "status": "0"
        }
      ]
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "roleids": [
      "5"
    ]
  },
  "id": 1
}
```

Veja também

- [Regras de função](#)
- [Elemento de IU](#)
- [Módulo](#)

- **Ação**

Fonte

CRole::create() in ui/include/classes/api/services/CRole.php.

role.delete

Descrição

`object role.delete(array roleids)`

Este método permite excluir funções.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs das funções a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos papéis excluídos na propriedade `roleids`.

Exemplos

Excluindo várias funções de usuário

Exclua duas funções de usuário.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "role.delete",
  "params": [
    "4",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "roleids": [
      "4",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CRole::delete() in ui/include/classes/api/services/CRole.php.

role.get

Descrição

`integer/array role.get(object parameters)`

O método permite recuperar roles de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
roleids	string/array	Retornar apenas papéis com os IDs fornecidos.
selectRules	query	Retornar regras de função na propriedade rules .
selectUsers	query	Retornar regras de função na propriedade rules .
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: roleid , name . Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro **countOutput** foi usado.

Exemplos**Recuperando dados de função**

Recupere os dados da função "Função de superadministrador" e suas regras de acesso.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "role.get",
  "params": {
    "output": "extend",
    "selectRules": "extend",
    "roleids": "3"
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "roleid": "3",
```



```

"name": "Super admin role",
"type": "3",
"readonly": "1",
"rules": {
  "ui": [
    {
      "name": "monitoring.dashboard",
      "status": "1"
    },
    {
      "name": "monitoring.problems",
      "status": "1"
    },
    {
      "name": "monitoring.hosts",
      "status": "1"
    },
    {
      "name": "monitoring.latest_data",
      "status": "1"
    },
    {
      "name": "monitoring.maps",
      "status": "1"
    },
    {
      "name": "monitoring.services",
      "status": "1"
    },
    {
      "name": "inventory.overview",
      "status": "1"
    },
    {
      "name": "inventory.hosts",
      "status": "1"
    },
    {
      "name": "reports.availability_report",
      "status": "1"
    },
    {
      "name": "reports.top_triggers",
      "status": "1"
    },
    {
      "name": "monitoring.discovery",
      "status": "1"
    },
    {
      "name": "reports.notifications",
      "status": "1"
    },
    {
      "name": "reports.scheduled_reports",
      "status": "1"
    },
    {
      "name": "configuration.host_groups",
      "status": "1"
    }
  ]
}

```

```

        "name": "configuration.templates",
        "status": "1"
    },
    {
        "name": "configuration.hosts",
        "status": "1"
    },
    {
        "name": "configuration.maintenance",
        "status": "1"
    },
    {
        "name": "configuration.actions",
        "status": "1"
    },
    {
        "name": "configuration.discovery",
        "status": "1"
    },
    {
        "name": "reports.system_info",
        "status": "1"
    },
    {
        "name": "reports.audit",
        "status": "1"
    },
    {
        "name": "reports.action_log",
        "status": "1"
    },
    {
        "name": "configuration.event_correlation",
        "status": "1"
    },
    {
        "name": "administration.general",
        "status": "1"
    },
    {
        "name": "administration.proxies",
        "status": "1"
    },
    {
        "name": "administration.authentication",
        "status": "1"
    },
    {
        "name": "administration.user_groups",
        "status": "1"
    },
    {
        "name": "administration.user_roles",
        "status": "1"
    },
    {
        "name": "administration.users",
        "status": "1"
    },
    {
        "name": "administration.media_types",
        "status": "1"
    }

```

```

    },
    {
      "name": "administration.scripts",
      "status": "1"
    },
    {
      "name": "administration.queue",
      "status": "1"
    }
  ],
  "ui.default_access": "1",
  "services.read.mode": "1",
  "services.read.list": [],
  "services.read.tag": {
    "tag": "",
    "value": ""
  },
  "services.write.mode": "1",
  "services.write.list": [],
  "services.write.tag": {
    "tag": "",
    "value": ""
  },
  "modules": [],
  "modules.default_access": "1",
  "api.access": "1",
  "api.mode": "0",
  "api": [],
  "actions": [
    {
      "name": "edit_dashboards",
      "status": "1"
    },
    {
      "name": "edit_maps",
      "status": "1"
    },
    {
      "name": "acknowledge_problems",
      "status": "1"
    },
    {
      "name": "close_problems",
      "status": "1"
    },
    {
      "name": "change_severity",
      "status": "1"
    },
    {
      "name": "add_problem_comments",
      "status": "1"
    },
    {
      "name": "execute_scripts",
      "status": "1"
    },
    {
      "name": "manage_api_tokens",
      "status": "1"
    }
  ]
}

```

```

        "name": "edit_maintenance",
        "status": "1"
    },
    {
        "name": "manage_scheduled_reports",
        "status": "1"
    }
],
"actions.default_access": "1"
}
}
],
"id": 1
}
}

```

Veja também

- [Regras de função](#)
- [Usuário](#)

Fonte

CRole::get() in ui/include/classes/api/services/CRole.php.

role.update

Descrição

object role.update(object/array roles)

Este método permite atualizar as funções existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades da função a serem atualizadas.

A propriedade roleid deve ser definida para cada função, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de função padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
rules	array	Acesse rules para substituir as regras de acesso atuais atribuídas à função.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos papéis atualizados na propriedade roleids.

Exemplos

Desativando a capacidade de executar scripts

Atualize a função com o ID "5", desative a capacidade de executar scripts.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "role.update",
  "params": [
    {

```

```

        "roleid": "5",
        "rules": {
            "actions": [
                {
                    "name": "execute_scripts",
                    "status": "0"
                }
            ]
        }
    },
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}

```

Limitando o acesso à API

Atualize a função com o ID "5", negue chamar qualquer método "criar", "atualizar" ou "excluir".

Request:

```

{
    "jsonrpc": "2.0",
    "method": "role.update",
    "params": [
        {
            "roleid": "5",
            "rules": {
                "api.access": "1",
                "api.mode": "0",
                "api": ["*.create", "/*.update", "/*.delete"]
            }
        }
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "roleids": [
            "5"
        ]
    },
    "id": 1
}

```

Fonte

CRole::update() in ui/include/classes/api/services/CRole.php.

Grupo de anfitriões

Esta classe foi projetada para trabalhar com grupos de hosts.

Referências de objetos:

- [Grupo de hosts](#)

Métodos disponíveis:

- [hostgroup.create](#) - criando novos grupos de hosts
- [hostgroup.delete](#) - excluindo grupos de hosts
- [hostgroup.get](#) - recuperando grupos de hosts
- [hostgroup.massadd](#) - adicionando objetos relacionados a grupos de hosts
- [hostgroup.massremove](#) - removendo objetos relacionados de grupos de hosts
- [hostgroup.massupdate](#) - substituindo ou removendo objetos relacionados de grupos de hosts
- [hostgroup.update](#) - atualizando grupos de hosts

> Objeto do grupo de hosts

Os objetos a seguir estão diretamente relacionados à API `hostgroup`.

Grupo de hosts

O objeto de grupo de hosts possui as seguintes propriedades.

Propriedade	Tipo	Descrição
groupid	string	(somente leitura) ID of the host group.
nome (obrigatório)	string	Nome do grupo de hosts.
flags	inteiro	(somente leitura) Origem do grupo de hosts. Valores possíveis: 0 - um grupo de hosts simples; 4 - um grupo de host de descoberta.
internal	inteiro	(somente leitura) Se o grupo é usado internamente pelo sistema. Um grupo interno não pode ser excluído. Valores possíveis: 0 - (padrão) não interno; 1 - interno.
uuid	string	Identificador universal exclusivo, usado para vincular grupos de hosts importados a grupos já existentes. Gerado automaticamente, se não fornecido.

Observe que para alguns métodos (atualização, exclusão), a combinação de parâmetros obrigatórios/opcionais é diferente.

`hostgroup.create`

Descrição

`object hostgroup.create(object/array hostGroups)`

Este método permite criar novos grupos de hosts.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(objeto/array) Grupos de hosts a serem criados. O método aceita grupos de hosts com as **propriedades padrão do grupo de hosts**.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host criado grupos sob a propriedade `groupids`. A ordem dos IDs retornados corresponde à ordem dos grupos de hosts passados.

Exemplos

Criando um grupo de hosts

Crie um grupo de hosts chamado "servidores Linux".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostgroup.create",
  "parâmetros": {
    "name": "Servidores Linux"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "groupids": [
      "107819"
    ]
  },
  "id": 1
}
```

Fonte

`CHostGroup::create()` em `ui/include/classes/api/services/CHostGroup.php`.

`hostgroup.delete`

Descrição

`object hostgroup.delete(array hostGroupIds)`

Este método permite excluir grupos de hosts.

Um grupo de hosts não pode ser excluído se:

- contém hosts que pertencem apenas a este grupo;
- está marcado como interno;
- é usado por um protótipo host;
- é usado em um script global;
- é usado em uma condição de correlação.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja **Usuário funções** Para maiores informações.

Parâmetros

(array) IDs dos grupos de hosts a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host excluído grupos sob a propriedade `groupids`.

Exemplos

Excluindo vários grupos de hosts

Exclua dois grupos de hosts.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostgroup.delete",
  "parâmetros": [
    "107824",
    "107825"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "groupids": [
      "107824",
      "107825"
    ]
  },
  "id": 1
}
```

Fonte

CHostGroup::delete() in ui/include/classes/api/services/CHostGroup.php.

hostgroup.get

Descrição

integer/array hostgroup.get(object parameters)

O método permite recuperar grupos de hosts de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro|**Tipo**|Descrição] |-----|-----| |graphids|string/array|Retorna apenas grupos de hosts que contêm hosts ou modelos com os gráficos fornecidos.| |groupids|string/array|Retorna apenas grupos de hosts com os IDs de grupos de hosts fornecidos.| |hostids|string/array|Retorna apenas grupos de hosts que contêm os hosts fornecidos.| |maintenanceids|string/array|Retorna apenas os grupos de hosts afetados pelas manutenções fornecidas.| |monitored_hosts|flag|Retorna apenas grupos de hosts que contêm hosts monitorados.| |real_hosts|flag|Retorna apenas grupos de hosts que contêm hosts.| |templated_hosts|flag|Retorna apenas grupos de hosts que contêm templates.| |templateids|string/array|Retorna apenas grupos de hosts que contêm os modelos fornecidos.| |triggerids|string/array|Retorna apenas grupos de hosts que contêm hosts ou modelos com os gatilhos fornecidos.| |with_graphs|flag|Retorna apenas grupos de hosts que contêm hosts com gráficos.| |with_graph_prototypes|flag|Retorna apenas grupos de hosts que contêm hosts com protótipos de gráficos.| |with_hosts_and_templates|flag|Retorne apenas grupos de hosts que contenham hosts ou templates.| |with_httptests|flag|Retorna apenas grupos de hosts que contêm hosts com verificações da web.

Substitui o parâmetro with_monitored_httptests.| |with_items|flag|Retorna apenas grupos de hosts que contenham hosts ou templates com itens.

Substitui os parâmetros with_monitored_items e with_simple_graph_items.| |with_item_prototypes|flag|Retorna

apenas grupos de hosts que contêm hosts com protótipos de itens.

Substitui o parâmetro `with_simple_graph_item_prototype` | `with_simple_graph_item_prototypes` | `flag` | Retorna apenas grupos de hosts que contêm hosts com protótipos de itens, que são ativados para criação e possuem informações do tipo numérico. | `with_monitored_httptests` | `flag` | Retorna apenas grupos de hosts que contenham hosts com verificações da web habilitadas. | `with_monitored_items` | `flag` | Retorna apenas grupos de hosts que contenham hosts ou templates com itens habilitados.

Substitui o parâmetro `with_simple_graph_items`. | `with_monitored_triggers` | `flag` | Retorna apenas grupos de hosts que contêm hosts com acionadores ativados. Todos os itens usados no gatilho também devem estar habilitados. | `with_simple_graph_items` | `flag` | Retorna apenas grupos de hosts que contenham hosts com itens numéricos. | `with_triggers` | `flag` | Retorna apenas grupos de hosts que contêm hosts com gatilhos.

Substitui o parâmetro `with_monitored_triggers`. | `selectDiscoveryRule` | `query` | Retorne uma propriedade `discoveryRule` com a regra LLD que criou o grupo de hosts. | `selectGroupDiscovery` | `query` | Retorne uma propriedade `groupDiscovery` com o objeto de descoberta do grupo de hosts.

O objeto de descoberta do grupo de hosts vincula um grupo de hosts descoberto a um protótipo de grupo de hosts e tem as seguintes propriedades:
`groupid` - (string) ID do grupo de hosts descoberto;
`lastcheck` - (timestamp) hora em que o grupo de hosts foi descoberto pela última vez;
`name` - (string) nome do protótipo do grupo de hosts;
`parent_group_prototypeid` - (string) ID do protótipo do grupo de hosts a partir do qual o grupo de hosts foi criado;
`ts_delete` - (timestamp) hora em que um grupo de hosts que é não mais descoberto será excluído. | `selectHosts` | `query` | Retorne uma propriedade `hosts` com os hosts que pertencem ao grupo de hosts.

Suporta `count`. | `selectTemplates` | `query` | Retorne uma propriedade `templates` com os templates que pertencem ao grupo de hosts.

Suporta `count`. | `limitSelects` | `integer` | Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:
`selectHosts` - os resultados serão classificados por host;
`selectTemplates` - resultados será ordenado por host. | `sortfield` | `string/array` | Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: `groupid`, `name`. | `countOutput` | `boolean` | Esses parâmetros sendo comuns para todos os métodos `get` são descritos em detalhes na página [comentário de referência](#). | `editável` | `boolean` | `excludePesquisa` | `boolean` | `filtro` | `objeto` | `limite` | `inteiro` | `saída` | `consulta` | `preservekeys` | `boolean` | `pesquisar` | `objeto` | `searchByAny` | `boolean` | `searchWildcardsEnabled` | `boolean` | `ordem de classificação` | `cadeia/matriz` | `startSearch` | `boolean` |

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro `countOutput` tiver sido usado.

Exemplos

Recuperando dados por nome

Recupere todos os dados sobre dois grupos de hosts chamados "servidores Zabbix" e "Servidores Linux".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostgroup.get",
  "parâmetros": {
    "saída": "estender",
    "filtro": {
      "nome": [
        "servidores Zabbix",
        "Servidores Linux"
      ]
    }
  },
  "auth": "6f38cddc44cfbb6c1bd186f9a220b5a0",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "groupid": "2",
      "name": "Servidores Linux",
      "interno": "0"
    },
    {
      "groupid": "4",
```

```

        "name": "servidores Zabbix",
        "interno": "0"
    }
],
    "id": 1
}

```

Veja também

- [Host](#)
- [Modelo](#)

Fonte

CHostGroup::get() em ui/include/classes/api/services/CHostGroup.php.

hostgroup.massadd

Descrição

`object hostgroup.massadd(object parameters)`

Este método permite adicionar simultaneamente vários objetos relacionados a todos os grupos de hosts fornecidos.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object) Todos os parâmetros de identificação dos grupos de hosts foram atualizados e os objetos a serem adicionados a todos os grupos de hosts.

O método aceita os seguintes parâmetros.

Parâmetros	Tipo	Descrição
groups (requerido)	object/array	Grupos de hosts a serem atualizados.
hosts	object/array	Os grupos de hosts devem ter a propriedade <code>groupid</code> definida. Hosts para adicionar a todos os grupos de hosts.
templates	object/array	Os hosts devem ter a propriedade <code>hostid</code> definida. Modelos para adicionar a todos os grupos de hosts.
		Os templates devem ter a propriedade <code>templateid</code> definida.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos grupos de hosts atualizados na propriedade `groupids`.

Exemplos

Adicionando hosts a grupos de hosts

Adicione dois hosts a grupos de hosts com IDs 5 e 6.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "hostgroup.massadd",
    "params": {
        "groups": [
            {
                "groupid": "5"
            },
            {

```

```

        "groupid": "6"
    },
    ],
    "hosts": [
        {
            "hostid": "30050"
        },
        {
            "hostid": "30001"
        }
    ]
},
"auth": "f223adf833b2bf2ff38574a67bba6372",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "groupids": [
            "5",
            "6"
        ]
    },
    "id": 1
}

```

Veja também

- [Host](#)
- [Template](#)

Fonte

CHostGroup::massAdd() in ui/include/classes/api/services/CHostGroup.php.

hostgroup.massremove

Descrição

object hostgroup.massremove(object parameters)

Este método permite remover objetos relacionados de vários grupos de hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo os IDs dos grupos de hosts a serem atualizados e os objetos que devem ser removidos.

Parâmetro	Tipo	Descrição
groupids	(obrigatório) string/array	IDs dos grupos de hosts a serem atualizados.
hostids	string/array	Hosts a serem removidos de todos os grupos de hosts.
templateids	string/array	Modelos a serem removidos de todos os grupos de hosts.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host atualizado grupos sob a propriedade groupids.

Exemplos

Removendo hosts de grupos de hosts

Remova dois hosts dos grupos de hosts fornecidos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "hostgroup.massremove",
  "parâmetros": {
    "groupids": [
      "5",
      "6"
    ],
    "hostids": [
      "30050",
      "30001"
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "groupids": [
      "5",
      "6"
    ]
  },
  "id": 1
}
```

Fonte

CHostGroup::massRemove() in ui/include/classes/api/services/CHostGroup.php.

hostgroup.massupdate

Descrição

`object hostgroup.massupdate(object parameters)`

Este método permite substituir hosts e templates pelos especificados aqueles em vários grupos de hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo os IDs dos grupos de hosts a serem atualizados e os objetos que devem ser atualizados.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **grupos**
(obrigatório)|objeto/array|Grupos de hosts a serem atualizados.

Os grupos de hosts devem ter a propriedade `groupid` definida.| **hosts**
(obrigatório)|object/array|Hosts para substituir os hosts atuais nos grupos de hosts fornecidos.
Todos os outros hosts, exceto os mencionados, serão excluídos dos grupos de hosts.
>Os hosts descobertos não serão afetados.

Os hosts devem ter a propriedade `hostid` definida.| **modelos**
(obrigatório)|objeto/array|Modelos para substituir os modelos atuais nos grupos de hosts especificados.
Todos os outros modelos, exceto os mencionados, serão excluídos dos grupos de hosts.
>
Os templates devem ter a propriedade `templateid` definida.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host atualizado grupos sob a propriedade `groupids`.

Exemplos

Substituindo hosts e modelos em um grupo de hosts

Substitua todos os hosts em um grupo de hosts pelos hosts mencionados e desvincule todos modelos do grupo de hosts.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "hostgroup.massupdate",
  "parâmetros": {
    "grupos": [
      {
        "groupid": "6"
      }
    ],
    "anfitriões": [
      {
        "hostid": "30050"
      }
    ],
    "modelos": []
  },
  "auth": "f223adf833b2bf2ff38574a67bba6372",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "groupids": [
      "6",
    ]
  },
  "id": 1
}
```

Veja também

- [hostgroup.update](#)
- [hostgroup.massadd](#)
- [Host](#)
- [Modelo](#)

Fonte

CHostGroup::massUpdate() em ui/include/classes/api/services/CHostGroup.php.

hostgroup.update

Descrição

object hostgroup.update(object/array hostGroups)

Este método permite atualizar grupos de hosts existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) **Propriedades do grupo de hosts** a ser Atualizada.

A propriedade groupid deve ser definida para cada grupo de hosts, todos os outros propriedades são opcionais. Apenas as propriedades fornecidas serão atualizadas, todas outros permanecerão inalterados.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host atualizado grupos sob a propriedade groupids.

Exemplos

Renomeando um grupo de hosts

Renomeie um grupo de hosts para "hosts Linux".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostgroup.update",
  "parâmetros": {
    "groupid": "7",
    "name": "Hospedeiros Linux"
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "groupids": [
      "7"
    ]
  },
  "id": 1
}
```

Fonte

CHostGroup::update() em ui/include/classes/api/services/CHostGroup.php.

Grupo de usuários

Esta classe foi projetada para trabalhar com grupos de usuários.

Referências de objetos:

- [Grupo de usuários](#)

Métodos disponíveis:

- [usergroup.create](#) - criando novos grupos de usuários
- [usergroup.delete](#) - excluindo grupos de usuários
- [usergroup.get](#) - recuperando grupos de usuários
- [usergroup.update](#) - atualizando grupos de usuários

> Objeto de grupo de usuários

Os objetos a seguir estão diretamente relacionados à API usergroup.

Grupo de usuários

O objeto de grupo de usuários tem as seguintes propriedades.

Propriedade	Tipo	Descrição
usrgrpId	string	(somente leitura) ID do grupo de usuários.
name (requerido)	string	Nome do grupo de usuários.
debug_mode	integer	Se o modo de depuração está ativado ou desativado.
Os valores possíveis são: 0 - (padrão) desativado; 1 - ativado.		

Propriedade	Tipo	Descrição
gui_access	integer	Método de autenticação de frontend dos usuários do grupo. Valores possíveis: 0 - (padrão) usar o método de autenticação padrão do sistema; 1 - usar interno autenticação; 2 - usar autenticação LDAP; 3 - desabilitar o acesso ao frontend.
users_status	integer	Se o grupo de usuários está habilitado ou desabilitado. Os valores possíveis são: 0 - (padrão) habilitado; 1 - desabilitado.

Permissão

O objeto de permissão tem as seguintes propriedades.

Propriedade	Tipo	Descrição
id (requerido)	string	ID do grupo de hosts ao qual adicionar permissão.
permission (requerido)	integer	Nível de acesso ao grupo de hosts. Valores possíveis: 0 - acesso negado; 2 - acesso somente leitura; 3 - acesso de leitura e gravação.

Permissão baseada em etiquetas

O objeto de permissão baseado em etiqueta tem as seguintes propriedades.

Propriedade	Tipo	Descrição
groupid (requerido)	string	ID do grupo de hosts ao qual adicionar permissão.
tag	string	Nome da etiqueta.
valor	string	Valor da etiqueta.

usergroup.create

Descrição

`object usergroup.create(object/array userGroups)`

Este método permite criar novos grupos de usuários.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parameters

(object/array) Grupos de usuários a serem criados.

Além dos [standard user group properties](#), o método aceita os seguintes parâmetros. ##### Parâmetros

(object/array) Grupos de usuários a serem criados.

Além das [propriedades do grupo de usuários padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
rights	object/array	Permissões para atribuir ao grupo
tag_filters	array	Permissões baseadas em etiquetas para atribuir ao grupo
users	object/array	Usuários para adicionar ao grupo de usuários.
Parameter	Type	O usuário deve ter a propriedade userid definida. Description
--	--	-----
rights	object/array	Permissions to assign to the group
tag_filters	array	Tag based permissions to assign to the group
users	object/array	Users to add to the user group.
		The user must have the userid property defined.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos grupos de usuários criados sob a propriedade `usrgrpids`. A ordem dos IDs retornados corresponde à ordem dos grupos de usuários passados.

Exemplos

Criando um grupo de usuários

Crie um grupo de usuários, que negue o acesso ao grupo de hosts "2", e adicione um usuário a ele.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usergroup.create",
  "params": {
    "name": "Operation managers",
    "rights": {
      "permission": 0,
      "id": "2"
    },
    "users": [
      {"userid": "12"}
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "usrgrpids": [
      "20"
    ]
  },
  "id": 1
}
```

Veja também

- [Permissão](#)

Fonte

`CUserGroup::create()` in `ui/include/classes/api/services/CUserGroup.php`.

usergroup.delete

Descrição

`object usergroup.delete(array userGroupIds)`

Este método permite excluir grupos de usuários.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário (user roles). Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos grupos de usuários a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos grupos de usuários excluídos sob a propriedade `usrgrpids`.

Exemplos

Excluindo vários grupos de usuários

Excluir dois grupos de usuários.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usergroup.delete",
  "params": [
    "20",
    "21"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "usrgrpids": [
      "20",
      "21"
    ]
  },
  "id": 1
}
```

Fonte

`CUserGroup::delete()` in `ui/include/classes/api/services/CUserGroup.php`.

usergroup.get

Descrição

`integer/array usergroup.get(object parameters)`

O método permite recuperar grupos de usuários de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
status	integer	Retorne apenas grupos de usuários com o status fornecido. Consulte a página do grupo de usuários para obter uma lista de status suportados.
userids	string/array	Retorna apenas grupos de usuários que contêm os usuários fornecidos.
usrgrpids	string/array	Retorna apenas grupos de usuários com os IDs fornecidos.
selectTagFilters	query	Retorna permissões baseadas em etiquetas de grupo de usuários na propriedade tag_filters . Tem as seguintes propriedades: groupid - (string) ID do grupo de hosts; tag - (string) nome da etiqueta; value - (string) valor da etiqueta.
selectUsers	query	Retorne os usuários do grupo de usuários na propriedade users .
selectRights	query	Retorne direitos de grupo de usuários em propriedade rights . Possui as seguintes propriedades: permission - nível de acesso (inteiro) ao grupo de hosts; id - (string) ID do grupo de hosts.
limitSelects	integer	Consulte a página do grupo de usuários para obter uma lista de níveis de acesso aos grupos de hosts.
sortfield	string/array	Limite o número de registros retornados por subseleções. Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: <code>usrgrpId</code> , <code>name</code> . Esses parâmetros sendo comuns para todos os métodos <code>get</code> são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma array de objetos;
- a contagem de objetos recuperados, se o parâmetro `countOutput` foi usado.

Exemplos

Recuperando grupos de usuários habilitados

Recuperar todos os grupos de usuários habilitados.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usergroup.get",
  "params": {
    "output": "extend",
    "status": 0
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
```

```
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "usrgrpid": "7",
      "name": "Zabbix administrators",
      "gui_access": "0",
      "users_status": "0",
      "debug_mode": "1"
    },
    {
      "usrgrpid": "8",
      "name": "Guests",
      "gui_access": "0",
      "users_status": "0",
      "debug_mode": "0"
    },
    {
      "usrgrpid": "11",
      "name": "Enabled debug mode",
      "gui_access": "0",
      "users_status": "0",
      "debug_mode": "1"
    },
    {
      "usrgrpid": "12",
      "name": "No access to the frontend",
      "gui_access": "2",
      "users_status": "0",
      "debug_mode": "0"
    },
    {
      "usrgrpid": "14",
      "name": "Read only",
      "gui_access": "0",
      "users_status": "0",
      "debug_mode": "0"
    },
    {
      "usrgrpid": "18",
      "name": "Deny",
      "gui_access": "0",
      "users_status": "0",
      "debug_mode": "0"
    }
  ],
  "id": 1
}
```

Veja também

- [Usuário](#)

Fonte

CUserGroup::get() in ui/include/classes/api/services/CUserGroup.php.

usergroup.update

Descrição

`object usergroup.update(object/array userGroups)`

Este método permite atualizar grupos de usuários existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do grupo de usuários a serem atualizadas.

A propriedade `usrgrp_id` deve ser definida para cada grupo de usuários, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades do grupo de usuários padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
<code>rights</code>	object/array	Permissões para substituir as permissões atuais atribuídas ao grupo de usuários.
<code>tag_filters</code>	array	Permissões baseadas em etiquetas para atribuir ao grupo.
<code>users</code>	object/array	Usuários para adicionar ao grupo de usuários.
O usuário deve ter a propriedade <code>userid</code> definida.		

Valores de retorno

(object) Retorna um objeto contendo os IDs dos grupos de usuários atualizados sob a propriedade `usrgrpids`.

Exemplos

Desativando um grupo de usuários

Desabilitar um grupo de usuários.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usergroup.update",
  "params": {
    "usrgrp_id": "17",
    "users_status": "1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "usrgrpids": [
      "17"
    ]
  },
  "id": 1
}
```

Veja também

- [Permissão](#)

Fonte

`CUserGroup::update()` in `ui/include/classes/api/services/CUserGroup.php`.

Gráfico

Esta classe foi projetada para trabalhar com itens.

Referências de objetos:

- [Gráfico](#)

Métodos disponíveis:

- [graph.create](#) - criando novo gráficos
- [graph.delete](#) - excluindo gráficos
- [graph.get](#) - recuperando gráficos
- [graph.update](#) - atualizando gráficos

> Objeto gráfico

Os objetos a seguir estão diretamente relacionados à API graph.

Gráfico

O objeto gráfico tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----|-----| |graphid|string|(readonly) ID do gráfico.|
|**altura**
(obrigatório)|inteiro|Altura do gráfico em pixels.| |**nome**
(obrigatório)|string|Nome do gráfico| |**largura**
(obrigatório)|inteiro|Largura do gráfico em pixels.| |flags|inteiro|(somente leitura) Origem do gráfico.

Os valores possíveis são:
0 - (padrão) um gráfico simples;
4 - um gráfico descoberto.| |graphtype|inteiro|Tipo de layout do gráfico.

Valores possíveis:
0 - (padrão) normal;
1 - empilhado;
2 - pizza;
3 - explodido .| |percent_left|float|Percentil esquerdo.

Padrão: 0.| |percent_right|float|Percentil direito.

Padrão: 0.| |show_3d|inteiro|Mostrar gráficos de pizza e explodidos em 3D.

Valores possíveis:
0 - (padrão) mostrar em 2D;
1 - mostrar em 3D.| |show_legend|integer|Se mostrar a legenda no gráfico.

Valores possíveis:
0 - ocultar;
1 - (padrão) mostrar.| |show_work_period|integer|Se mostrar o tempo de trabalho no gráfico.

Valores possíveis:
0 - ocultar;
1 - (padrão) mostrar.| |show_triggers|integer|Se mostrar a linha de disparo no gráfico.

Valores possíveis:
0 - ocultar;
1 - (padrão) mostrar.| |templateid|string|(readonly) ID do gráfico de modelo pai.| |yaxismax|float|O valor máximo fixo para o eixo Y.

Padrão: 100.| |yaxismin|float|O valor mínimo fixo para o eixo Y.

Padrão: 0.| |ymax_itemid|string|ID do item que é usado como o valor máximo para o eixo Y.| |ymax_type|inteiro|Método de cálculo do valor máximo para o eixo Y.

Valores possíveis:
0 - (padrão) calculado;
1 - fixo;
2 - item .| |ymin_itemid|string|ID do item que é usado como valor mínimo para o eixo Y.| |ymin_type|inteiro|Método de cálculo do valor mínimo para o eixo Y.

Valores possíveis:
0 - (padrão) calculado;
1 - fixo;
2 - item .| |uuid|string|Identificador único universal, usado para vincular gráficos importados a gráficos já existentes. Usado apenas para gráficos em modelos. Gerado automaticamente, se não fornecido.

Para operações de atualização, este campo é somente leitura.|

graph.create

Descrição

object graph.create(object/array graphs)

Este método permite criar novos gráficos.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Gráficos para criar.

Além das **propriedades de gráfico padrão**, o O método aceita os seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----|-----| |**gitems**
(obrigatório)|array|Gráfico **itens** a ser criado para o gráfico.|

Valores de retorno

(object) Retorna um objeto contendo os IDs dos gráficos criados sob a propriedade `graphids`. A ordem dos IDs retornados corresponde à ordem dos gráficos passados.

Exemplos

Criando um gráfico

Crie um gráfico com dois itens.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "graf.criar",
  "parâmetros": {
    "name": "Largura de banda MySQL",
    "largura": 900,
    "altura": 200,
    "gitens": [
      {
        "itemid": "22828",
        "cor": "00AA00"
      },
      {
        "itemid": "22829",
        "cor": "3333FF"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "grafos": [
      "652"
    ]
  },
  "id": 1
}
```

Veja também

- [Item de gráfico](#)

Fonte

CGraph::create() em `ui/include/classes/api/services/CGraph.php`.

graph.delete

Descrição

object graph.delete(array graphIds)

Este método permite excluir gráficos.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos gráficos a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos gráficos excluídos sob a propriedade `graphids`.

Exemplos

Excluindo vários gráficos

Exclua dois gráficos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "graph.delete",
  "parâmetros": [
    "652",
    "653"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "grafos": [
      "652",
      "653"
    ]
  },
  "id": 1
}
```

Fonte

CGraph::delete() em `ui/include/classes/api/services/CGraph.php`.

`graph.get`

Descrição

`integer/array graph.get(object parameters)`

O método permite recuperar gráficos de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro|**Tipo**|Descrição] |-----|-----| |graphids|string/array|Retorna apenas gráficos com os IDs fornecidos.| |groupids|string/array|Retorna apenas gráficos que pertencem a hosts nos grupos de hosts especificados.| |templateids|string/array|Retorna apenas o gráfico que pertence aos modelos fornecidos.| |hostids|string/array|Retorna apenas os gráficos que pertencem aos hosts fornecidos.| |itemids|string/array|Retorna apenas os gráficos que contêm os itens fornecidos.| |templated|boolean|Se definido como `true`, retornará apenas os gráficos que pertencem aos modelos.| |inherited|boolean|Se definido como `true`, retornará apenas os gráficos herdados de um modelo.| |expandName|flag|Expandir macros no nome do gráfico.| |selectGroups|query|Retorne uma propriedade **groups** com os grupos de hosts aos quais o gráfico pertence.| |selectTemplates|query|Retorne uma propriedade **templates** com os templates aos quais o gráfico pertence.| |selectHosts|query|Retorne uma propriedade **hosts** com os hosts aos quais o gráfico pertence.| |selectItems|query|Retorne uma propriedade **items** com os itens usados no gráfico.| |selectGraphDiscovery|query|Retorne uma propriedade **graphDiscovery** com o objeto de descoberta de gráfico.

Os objetos de descoberta de gráfico vinculam o gráfico a um protótipo de gráfico a partir do qual ele foi criado.

Ele tem as seguintes propriedades:
graphid - (string) ID do gráfico;
parent_graphid - (string) ID do protótipo do gráfico a partir do qual o gráfico foi criado.| |selectGraphItems|query|Retorne uma propriedade **gitems** com os itens usados no gráfico.| |selectDiscoveryRule|query|Retorne uma propriedade **discoveryRule** com a regra de descoberta de baixo nível que criou o gráfico.| |filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.

Suporta filtros adicionais:
host - nome técnico do host ao qual o gráfico pertence;
hostid - ID do host ao qual o gráfico pertence.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: graphid, name e graphtype.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página **comentário de referência**.| |editável|boolean|^| |excludePesquisa|boolean|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando gráficos de hosts

Recupere todos os gráficos do host "10107" e classifique-os por nome.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "graf.get",
  "parâmetros": {
    "saída": "estender",
    "hostids": 10107,
    "sortfield": "nome"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "grafid": "612",
      "name": "CPU salta",
      "largura": "900",
      "altura": "200",
      "yaxismin": "0",
      "yaxismax": "100",
      "modeloid": "439",
      "show_work_period": "1",
      "show_triggers": "1",
      "tipo de gráfico": "0",
      "show_legend": "1",
      "show_3d": "0",
      "percent_left": "0",
      "percent_right": "0",
      "ymin_type": "0",
      "ymax_type": "0",
      "ymin_itemid": "0",
      "ymax_itemid": "0",
      "bandeiras": "0"
    },
    {
      "grafid": "613",
```



```

        "name": "Carga da CPU",
        "largura": "900",
        "altura": "200",
        "yaxismin": "0",
        "yaxismax": "100",
        "templateid": "433",
        "show_work_period": "1",
        "show_triggers": "1",
        "tipo de gráfico": "0",
        "show_legend": "1",
        "show_3d": "0",
        "percent_left": "0",
        "percent_right": "0",
        "ymin_type": "1",
        "ymax_type": "0",
        "ymin_itemid": "0",
        "ymax_itemid": "0",
        "bandeiras": "0"
    },
    {
        "grafid": "614",
        "name": "Utilização da CPU",
        "largura": "900",
        "altura": "200",
        "yaxismin": "0",
        "yaxismax": "100",
        "templateid": "387",
        "show_work_period": "1",
        "show_triggers": "0",
        "graphtype": "1",
        "show_legend": "1",
        "show_3d": "0",
        "percent_left": "0",
        "percent_right": "0",
        "ymin_type": "1",
        "ymax_type": "1",
        "ymin_itemid": "0",
        "ymax_itemid": "0",
        "bandeiras": "0"
    },
    {
        "grafid": "645",
        "name": "Uso do espaço em disco /",
        "largura": "600",
        "altura": "340",
        "yaxismin": "0",
        "yaxismax": "0",
        "templateid": "0",
        "show_work_period": "0",
        "show_triggers": "0",
        "graphtype": "2",
        "show_legend": "1",
        "show_3d": "1",
        "percent_left": "0",
        "percent_right": "0",
        "ymin_type": "0",
        "ymax_type": "0",
        "ymin_itemid": "0",
        "ymax_itemid": "0",
        "bandeiras": "4"
    }
],

```

```
    "id": 1
}
```

Veja também

- [Descoberta regra](#)
- [Item de gráfico](#)
- [Item](#)
- [Host](#)
- [Grupo de hosts](#)
- [Modelo](#)

Fonte

CGraph::get() em ui/include/classes/api/services/CGraph.php.

graph.update

Descrição

object graph.update(object/array graphs)

Este método permite atualizar os gráficos existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do gráfico a serem atualizadas.

A propriedade graphid deve ser definida para cada gráfico, todos os outros propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas os outros permanecerão inalterados.

Além das [propriedades de gráfico padrão](#), o método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [gitems|array|Gráfico items para substituir itens de gráfico existentes. Se um item de gráfico tiver a propriedade gitemid definida, ele será atualizado, caso contrário, um novo item de gráfico será criado.]

Valores de retorno

(object) Retorna um objeto contendo os IDs dos gráficos atualizados sob a propriedade graphids.

Exemplos

Configurando o máximo para a escala Y

Defina o máximo da escala Y para um valor fixo de 100.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "graph.update",
  "parâmetros": {
    "grafid": "439",
    "ymax_type": 1,
    "yaxismax": 100
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "grafos": [
```

```

    ],
    "id": 1
}

```

Fonte

CGraph::update() em ui/include/classes/api/services/CGraph.php.

História

Esta classe foi projetada para trabalhar com dados de histórico.

Referências de objetos:

- [Histórico](#)

Métodos disponíveis:

- [history.get](#) - recuperando dados do histórico.

> Objeto histórico

Os objetos a seguir estão diretamente relacionados à API `history`.

Note:

Os objetos de histórico diferem dependendo do tipo de formação. Eles são criados pelo servidor Zabbix e não podem ser modificados por meio da API.

Histórico de flutuação

O objeto de histórico de flutuação tem as seguintes propriedades.

Propriedade	Tipo	Descrição
clock timestamp		Horário em que esse valor foi recebido.
itemid string		ID do item relacionado.
ns inteiro		Nanosegundos quando o valor foi recebido.
valor float		Valor recebido.

Histórico inteiro

O objeto de histórico inteiro tem as seguintes propriedades.

Propriedade	Tipo	Descrição
clock timestamp		Horário em que esse valor foi recebido.
itemid string		ID do item relacionado.
ns inteiro		Nanosegundos quando o valor foi recebido.
valor inteiro		Valor recebido.

Histórico de strings

O objeto de histórico de strings tem as seguintes propriedades.

Propriedade	Tipo	Descrição
clock timestamp		Horário em que esse valor foi recebido.
itemid string		ID do item relacionado.
ns inteiro		Nanosegundos quando o valor foi recebido.
valor string		Valor recebido.

Histórico de texto

O objeto de histórico de texto tem as seguintes propriedades.

Propriedade	Tipo	Descrição
id string		ID da entrada do histórico.
clock timestamp		Horário em que esse valor foi recebido.
itemid string		ID do item relacionado.
ns inteiro		Nanosegundos quando o valor foi recebido.
valor texto		Valor recebido.

Histórico de registros

O objeto de histórico de log tem as seguintes propriedades.

Propriedade	Tipo	Descrição
id string		ID da entrada do histórico.
clock timestamp		Horário em que esse valor foi recebido.
itemid string		ID do item relacionado.
logeventid integer		ID da entrada do log de eventos do Windows.
ns inteiro		Nanosegundos quando o valor foi recebido.
severity inteiro		Nível de entrada

do log de eventos do Windows.| |source|string|Fonte de entrada do log de eventos do Windows.| |timestamp|timestamp|Hora de entrada do log de eventos do Windows.| |valor|texto|Valor recebido.|

history.get

Descrição

integer/array history.get(object parameters)

O método permite recuperar histórico de dados de acordo com o parametros dados.

Veja também: [known issues](#)

Attention:

Este método pode retornar históricos de dados de uma entidade excluída se esses dados ainda não foram removidos definitivamente

Note:

Este metodo está disponível para qualquer tipo de usuário. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Veja [User roles](#) para mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| |history|integer|Tipos de objeto de histórico a serem retornados.

Valores possíveis:
0 - numérico float;
1 - caractere;
2 - log;
3 - numérico unsigned;
4 - texto.

Padrão: 3.| |hostids|string/array|Retorna apenas o histórico dos hosts fornecidos.| |itemids|string/array|Retorna apenas o histórico dos itens fornecidos.| |time_from|timestamp|Retorna apenas os valores que foram recebidos após ou no momento determinado.| |time_till|timestamp|Retorna apenas os valores que foram recebidos antes ou no momento determinado.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: itemid e clock.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página [comentário de referência](#).| |editável|boolean|^| |excluePesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando dados do histórico de itens

Retorna os 10 últimos valores recebidos de um item numérico (flutuante).

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "história.get",
  "parâmetros": {
    "saída": "estender",
    "história": 0,
    "itemids": "23296",
    "sortfield": "relógio",
    "ordem de classificação": "DESC",
    "limite": 10
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "itemid": "23296",
      "relógio": "1351090996",
      "valor": "0,085",
      "ns": "563157632"
    },
    {
      "itemid": "23296",
      "relógio": "1351090936",
      "valor": "0,16",
      "ns": "549216402"
    },
    {
      "itemid": "23296",
      "relógio": "1351090876",
      "valor": "0,18",
      "ns": "537418114"
    },
    {
      "itemid": "23296",
      "relógio": "1351090816",
      "valor": "0,21",
      "ns": "522659528"
    },
    {
      "itemid": "23296",
      "relógio": "1351090756",
      "valor": "0,215",
      "ns": "507809457"
    },
    {
      "itemid": "23296",
      "relógio": "1351090696",
      "valor": "0,255",
      "ns": "495509699"
    },
    {
      "itemid": "23296",
      "relógio": "1351090636",
      "valor": "0,36",
      "ns": "477708209"
    },
    {
      "itemid": "23296",
      "relógio": "1351090576",
      "valor": "0,375",
      "ns": "463251343"
    },
    {
      "itemid": "23296",
      "relógio": "1351090516",
      "valor": "0,315",
      "ns": "447947017"
    },
    {
      "itemid": "23296",
      "relógio": "1351090456",
      "valor": "0,275",

```

```

        "ns": "435307141"
    }
],
    "id": 1
}

```

Fonte

CHistory::get() em ui/include/classes/api/services/CHistory.php.

histórico.limpar

Descrição

`object history.clear(array itemids)`

Este método permite limpar o histórico do item.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs de itens a serem limpos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos itens compensados sob a propriedade `itemids`.

Exemplos

Apagar o histórico

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "histórico.limpar",
    "parâmetros": [
        "10325",
        "13205"
    ],
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "items": [
            "10325",
            "13205"
        ]
    },
    "id": 1
}

```

Fonte

CHistory::clear() em ui/include/classes/api/services/CHistory.php.

Host

Esta classe foi projetada para trabalhar com hosts.

Referências de objetos:

- **Host**
- **Inventário do host**

Métodos disponíveis:

- **host.create** - criando novo anfitriões
- **host.delete** - excluindo hosts
- **host.get** - recuperando hosts
- **host.massadd** - adicionando relacionado objetos para hosts
- **host.massremove** - removendo objetos relacionados de hosts
- **host.massupdate** - substituindo ou removendo objetos relacionados dos hosts
- **host.update** - atualizando hosts

> Objeto host

Os objetos a seguir estão diretamente relacionados à API do host.

Hospedeiro

O objeto host tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |hostid|string|(readonly) ID do host. |
|**host**
(obrigatório)|string|Nome técnico do host. | |descricao|texto|Descrição do host. | |flags|inteiro|(somente leitura) Origem do host.

Valores possíveis:
0 - um host simples;
4 - um host descoberto. | |inventario_modo|inteiro|Modo de preenchimento do inventário do host.

Os valores possíveis são:
-1 - (padrão) desativado;
0 - manual;
1 - automático. | |ipmi_authtype|inteiro|Algoritmo de autenticação IPMI.

Os valores possíveis são:
-1 - (padrão) padrão;
0 - nenhum;
1 - MD2;
2 - MD5
4 - reto;
5 - OEM;
6 - RMCP+. | |ipmi_password|string|senha do IPMI. | |ipmi_privilege|inteiro|Nível de privilégio IPMI.

Os valores possíveis são:
1 - retorno de chamada;
2 - (padrão) usuário;
3 - operador;
4 - admin;
5 - OEM. | |ipmi_username|string|IPMI username. | |manutencao_from|timestamp|(somente leitura) Hora de início da manutenção efetiva. | |manutencao_status|inteiro|(somente leitura) Status de manutenção efetiva.

Os valores possíveis são:
0 - (padrão) sem manutenção;
1 - manutenção em vigor. | |manutencao_tipo|inteiro|(somente leitura) Tipo de manutenção efetiva.

Os valores possíveis são:
0 - (padrão) manutenção com coleta de dados;
1 - manutenção sem coleta de dados. | |maintenaceid|string|(readonly) ID da manutenção que está atualmente em vigor no host. | |name|string|Nome visível do host.

Padrão: valor da propriedade host. | |proxy_hostid|string|ID do proxy que é usado para monitorar o host. | |status|inteiro|Status e função do host.

Os valores possíveis são:
0 - (padrão) host monitorado;
1 - host não monitorado. | |tls_connect|integer|Conexões ao host.

Os valores possíveis são:
1 - (padrão) Sem criptografia;
2 - PSK;
4 - certificado. | |tls_accept|inteiro|Conexões do host.

Os valores de bitmap possíveis são:
1 - (padrão) Sem criptografia;
2 - PSK;
4 - certificado. | |tls_issuer|string|Emissor do certificado. | |tls_subject|string|Assunto do certificado. | |tls_psk_identity|string|(somente gravação) identidade PSK. Necessário se tls_connect ou tls_accept tiver o PSK ativado.

Não coloque informações confidenciais na identidade do PSK, elas são transmitidas sem criptografia pela rede para informar ao receptor qual PSK usar. | |tls_psk|string|(write-only) A chave pré-compartilhada, com pelo menos 32 dígitos hexadecimais. Necessário se tls_connect ou tls_accept tiver o PSK ativado. |

Inventário de host

O objeto de inventário do host tem as seguintes propriedades.

Cada propriedade tem seu próprio número de identificação exclusivo, que é usado para associar campos de inventário de host a itens.

[ID|Propriedade|**Tipo**|Descrição] |--|-----|-----| |4|alias|string|Alias. | |11|ativo_tag|string|Etiqueta do ativo. | |28|chassis|string|chassis. | |23|contato|string|Pessoa de contato. | |32|contrato_numero|string|Número do contrato. | |47|data_hw_decomm|string|data de desativação do HW. | |46|data_hw_expiry|string|data de expiração da manutenção do HW. | |45|data_hw_install|string|data de instalação do HW. | |44|data_hw_purchase|string|data de compra do HW. | |34|implantacao_status|string|Status da implantação. | |14|hardware|string|Hardware. | |15|hardware_full|string|Hardware detalhado. | |39|host_netmask|string|Máscara de sub-rede do host. | |38|host_networks|string|Redes de host. | |40|host_router|string|Roteador host. | |30|hw_arch|string|arquitetura HW. | |33|instalador_name|string|Nome do instalador. | |24|localizacao|string|Localização. | |25|local_lat|string|Latitude do local. | |26|local_lon|string|Longitude do local. | |12|macaddress_a|string|endereço MAC A. | |13|macaddress_b|string|endereço MAC B. | |29|modelo|string|Modelo. | |3|nome|string|Nome. | |27|notas|cadeia|Notas. | |41|oob_ip|string|endereço IP OOB. | |42|oob_netmask|string|máscara de sub-rede do host OOB. | |43|oob_router|string|roteador OOB. | |5|os|string|Nome do SO. | |6|os_full|string|Nome detalhado do SO. | |7|os_short|string|Nome curto do SO. | |61|poc_1_cell|string|Número de celular POC principal. | |58|poc_1_email|string|E-mail principal. | |57|poc_1_name|string|Nome POC principal. | |63|poc_1_notes|string|Notas POC primárias. | |59|poc_1_phone_a|string|Telefone POC principal A. | |60|poc_1_phone_b|string|Telefone POC principal B. |

|62|poc_1_screen|string|Nome da tela POC principal.| |68|poc_2_cell|string|Número de celular POC secundário.| |65|poc_2_email|string|E-mail POC secundário.| |64|poc_2_name|string|Nome do POC secundário.| |70|poc_2_notes|string|Notas POC secundárias.| |66|poc_2_phone_a|string|Telefone POC secundário A.| |67|poc_2_phone_b|string|Telefone POC secundário B.| |69|poc_2_screen|string|Nome de tela POC secundário.| |8|serialno_a|string|Número de série A.| |9|serialno_b|string|Número de série B.| |48|site_address_a|string|Endereço do site A.| |49|site_address_b|string|Endereço do site B.| |50|site_address_c|string|Endereço do site C.| |51|site_city|string|Site city.| |53|site_country|string|País do site.| |56|site_notes|string|Notas do site.| |55|site_rack|string|Localização do rack do site.| |52|site_state|string|Estado do site.| |54|site_zip|string|CEP do site/código postal.| |16|software|string|Software.| |18|software_app_a|string|Aplicativo de software A.| |19|software_app_b|string|Aplicativo de software B.| |20|software_app_c|string|Aplicativo de software C.| |21|software_app_d|string|Aplicativo de software D.| |22|software_app_e|string|Aplicativo de software E.| |17|software_full|string|Detalhes do software.| |10|tag|string|Tag.| |1|tipo|string|Tipo.| |2|type_full|string|Digite detalhes.| |35|url_a|string|URL A.| |36|url_b|string|URL B.| |37|url_c|string|URL C.| |31|fornecedor|string|Fornecedor.|

Etiqueta do anfitrião

O objeto de tag de host possui as seguintes propriedades.

Propriedade	Tipo	Descrição
tag	 (obrigatório)	string Nome da tag do host.
valor		string Valor da tag do host.

host.create

Descrição

`object host.create(object/array hosts)`

Este método permite criar novos hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(objeto/array) Hosts para criar.

Além das **propriedades padrão do host**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
grupos	 (obrigatório)	objeto/matriz Host grupos ao qual adicionar o host. Os grupos de hosts devem ter a propriedade <code>groupid</code> definida.
interfaces	object/array	Interfaces a ser criado para o host.
tags	object/array	Host tags .
templates	object/array	Templates a serem vinculados ao host. Os templates devem ter a propriedade <code>templateid</code> definida.
macros	object/array	Macros do usuário a serem criados para o host.
inventário	objeto	Host inventário propriedades.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts criados sob a propriedade `hostids`. A ordem dos IDs retornados corresponde à ordem dos hosts passados.

Exemplos

Criando um host

Crie um host chamado "servidor Linux" com uma interface IP e tags, adicione a um grupo, vincule um modelo a ele e defina os endereços MAC no inventário do hospedeiro.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.create",
  "parâmetros": {
    "host": "Servidor Linux",
    "interfaces": [
      {
        "tipo": 1,
        "principal": 1,
        "useip": 1,
        "ip": "192.168.3.1",
        "dns": "",

```



```

        "porta": "10050"
    },
    ],
    "grupos": [
        {
            "groupid": "50"
        }
    ],
    "Tag": [
        {
            "tag": "Nome do host",
            "value": "Servidor Linux"
        }
    ],
    "modelos": [
        {
            "templateid": "20045"
        }
    ],
    "macros": [
        {
            "macro": "${USER_ID}",
            "valor": "123321"
        },
        {
            "macro": "${USER_LOCATION}",
            "valor": "0:0:0",
            "description": "coordenadas de latitude, longitude e altitude"
        }
    ],
    "modo_inventário": 0,
    "inventário": {
        "macaddress_a": "01234",
        "macaddress_b": "56768"
    }
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "hostids": [
            "107819"
        ]
    },
    "id": 1
}

```

Criando um host com interface SNMP

Crie um host chamado "host SNMP" com uma interface SNMPv3 com detalhes.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "host.create",
    "parâmetros": {
        "host": "Host SNMP",
        "interfaces": [
            {

```

```

        "tipo 2,
        "principal": 1,
        "useip": 1,
        "ip": "127.0.0.1",
        "dns": "",
        "porta": "161",
        "detalhes": {
            "versão": 3,
            "em massa": 0,
            "securityname": "mysecurityname",
            "contextname": "",
            "nível de segurança": 1
        }
    },
    ],
    "grupos": [
        {
            "groupid": "4"
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "hostids": [
            "10658"
        ]
    },
    "id": 1
}

```

Creating a host with PSK encryption

Create a host called "PSK host" with PSK encryption configured. Note that the host has to be **pre-configured to use PSK**.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "host.create",
    "params": {
        "host": "PSK host",
        "interfaces": [
            {
                "type": 1,
                "ip": "192.168.3.1",
                "dns": "",
                "port": "10050",
                "useip": 1,
                "main": 1
            }
        ],
        "groups": [
            {
                "groupid": "2"
            }
        ],
        "tls_accept": 2,
        "tls_connect": 2,
    }
}

```

```

    "tls_psk_identity": "PSK 001",
    "tls_psk": "1f87b595725ac58dd977beef14b97461a7c1045b9a1c963065002c5473194952"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "hostids": [
      "10590"
    ]
  },
  "id": 1
}

```

Veja também

- [Grupo de hosts](#)
- [Modelo](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)
- [Inventário do host](#)
- [Tag do host](#)

Fonte

CHost::create() em ui/include/classes/api/services/CHost.php.

host.delete

Descrição

`object host.delete(array hosts)`

Este método permite excluir hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs de hosts a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts excluídos sob a propriedade `hostids`.

Exemplos

Excluindo vários hosts

Exclua dois hosts.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "host.delete",
  "parâmetros": [
    "13",
    "32"
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
}

```

```

    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "hostids": [
            "13",
            "32"
        ]
    },
    "id": 1
}

```

Fonte

CHost::delete() em ui/include/classes/api/services/CHost.php.

host.get

Descrição

integer/array host.get(object parameters)

O método permite recuperar hosts de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro Tipo]	[Descrição]
[groupids]	string/array Retorna apenas hosts que pertencem aos grupos indicados.
[dserviceids]	string/array Retorna apenas hosts relacionados aos serviços descobertos fornecidos.
[graphids]	string/array Retorna apenas hosts que possuem os gráficos fornecidos.
[hostids]	string/array Retorna apenas hosts com os IDs de host fornecidos.
[httpstests]	string/array Retorna apenas hosts que possuem as verificações da web fornecidas.
[interfaceids]	string/array Retorna apenas hosts que usam as interfaces fornecidas.
[itemids]	string/array Retorna apenas hosts que possuem os itens fornecidos.
[maintenanceids]	string/array Retorna apenas os hosts afetados pelas manutenções fornecidas.
[monitored_hosts]	flag Retorna apenas hosts monitorados.
[proxy_hosts]	flag Retorna apenas proxies.
[proxyids]	string/array Retorna apenas os hosts monitorados pelos proxies fornecidos.
[templated_hosts]	flag Retorne tanto hosts quanto templates.
[templateids]	string/array Retorna apenas hosts que estão vinculados aos modelos fornecidos.
[triggerids]	string/array Retorna apenas hosts que possuem os gatilhos fornecidos.
[with_items]	flag Retorna apenas hosts que possuem itens.
 Substitui os parâmetros with_monitored_items e with_simple_graph_items.	
[with_item_prototypes]	flag Retorna apenas hosts que possuem protótipos de itens.
 Substitui o parâmetro with_simple_graph_items.	
[with_simple_graph_item_prototypes]	flag Retorna apenas hosts que possuem protótipos de itens, que são habilitados para criação e possuem tipo numérico de informação.
[with_graphs]	flag Retorna apenas hosts que possuem gráficos.
[with_graph_prototypes]	flag Retorna apenas hosts que possuem protótipos de grafos.
[with_httpstests]	flag Retorna apenas hosts que possuem verificações da web.
 Substitui o parâmetro with_monitored_httpstests.	
[with_monitored_httpstests]	flag Retorna apenas hosts que habilitaram verificações da web.
[with_monitored_items]	flag Retorna apenas hosts que tenham itens habilitados.
 Substitui o parâmetro with_simple_graph_items.	
[with_monitored_triggers]	flag Retorna apenas os hosts que possuem triggers habilitados. Todos os itens usados no gatilho também devem estar habilitados.
[with_simple_graph_items]	flag Retorna apenas hosts que possuem itens com tipo numérico de informação.
[with_triggers]	flag Retorna apenas hosts que possuem triggers.
 Substitui o parâmetro with_monitored_triggers.	
[withProblemsSuppressed]	boolean Retorna hosts que suprimiram problemas.
 Valores possíveis: null - (padrão) todos os hosts; true - somente hosts com problemas suprimidos; >false - somente hosts com problemas não suprimidos.	
[valtype]	integer Regras para pesquisa de tags.
 Valores possíveis: 0 - (padrão) E/Ou; 2 - Ou.	
[severities]	integer/array Anfitriões de retorno que têm apenas problemas com determinadas gravidades. Aplica-se apenas se o objeto do problema for acionador.
[tags]	array/object Retorna apenas hosts com as tags fornecidas. Correspondência exata por tag e pesquisa com distinção entre maiúsculas e minúsculas por valor de tag, dependendo do valor do operador.
 Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}], ...]. Um array vazio retorna todos os hosts.	
 Possíveis valores de operador: 0 - (padrão)	

Contém;
1 - Igual;
2 - Não gosto;
3 - Não igual
4 - Existe;
5 - Não existe.| |inheritedTags|boolean|Retorna hosts que deram tags também em todos os seus templates vinculados. Padrão:

Valores possíveis:
true - templates vinculados também devem ter dado tags;
false - (padrão) tags de templates vinculados são ignoradas.| |selectDiscoveries|query|Retorne uma propriedade **discoveries** com regras de descoberta de baixo nível do host.

Suporta count.| |selectDiscoveryRule|query|Retorne uma propriedade **discoveryRule** com a regra de descoberta de baixo nível que criou o host (do protótipo do host no monitoramento VMware).| |selectGraphs|query|Retorne uma propriedade **graphs** com gráficos de host.

Suporta count.| |selectGroups|query|Retorne uma propriedade **groups** com dados de grupos de hosts aos quais o host pertence.| |selectHostDiscovery|query|Retorne uma propriedade **hostDiscovery** com dados do objeto de descoberta de host.

O objeto de descoberta de host vincula um host descoberto a um protótipo de host ou protótipos de host a uma regra LLD e tem as seguintes propriedades:
host - (string) host do protótipo do host;
hostid - (string) ID do host ou protótipo do host descoberto;
parent_hostid - (string) ID do protótipo do host a partir do qual o host foi criado;
parent_itemid - (string) ID da regra LLD que criou o host descoberto;
lastcheck - (timestamp) hora em que o host foi descoberto pela última vez;
ts_delete - (timestamp) hora em que um host que não é longer descoberto será excluído.| |selectHttpTests|query|Retorne uma propriedade **httpTests** com cenários da web do host.

Suporta count.| |selectInterfaces|query|Retorna uma propriedade de **interfaces** com interfaces de host.

Suporta count.| |selectInventory|query|Retorne uma propriedade de **inventário** com dados de inventário do host.| |selectItems|query|Retorne uma propriedade **items** com itens do host.

Suporta count.| |selectMacros|query|Retorne uma propriedade **macros** com macros do host.| |selectParentTemplates|query|Retorne uma propriedade **parentTemplates** com modelos aos quais o host está vinculado.

Suporta count.| |selectDashboards|query|Retorna uma propriedade de **painéis**.

Suporta count.| |selectTags|query|Retorne uma propriedade **tags** com tags de host.| |selectInheritedTags|query|Retorne uma propriedade **inheritedTags** com tags que estão em todos os templates que estão vinculados ao host.| |selectTriggers|query|Retorne uma propriedade **triggers** com acionadores de host.

Suporta count.| |selectValueMaps|query|Retorne uma propriedade **valuemaps** com mapas de valor do host.| |filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.

Permite filtrar por propriedades da interface.| |limitSelects|integer|Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:
selectParentTemplates - os resultados serão classificados por host;
selectInterfaces;
selectItems - ordenado por name;
selectDiscoveries - ordenado por name;
selectTriggers - ordenado por description;
selectGraphs - ordenado por name;
selectDashboards - ordenado por name.| |pesquisa|objeto|Retorna resultados que correspondam à pesquisa de curinga fornecida.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são strings a serem pesquisadas. Se nenhuma opção adicional for fornecida, isso fará uma pesquisa LIKE "%...%".

Permite pesquisar pelas propriedades da interface. Funciona apenas com campos de texto.| |searchInventory|object|Retorna apenas hosts que tenham dados de inventário que correspondam à pesquisa de curinga fornecida.

Este parâmetro é afetado pelos mesmos parâmetros adicionais que search.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: hostid, host, name, status.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no **comentário de referência**.| |editável|booleano|^| |excludePesquisa|boolean|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando dados por nome

Recupere todos os dados sobre dois hosts chamados "servidor Zabbix" e "Linux servidor".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "filtro": {
      "hospedeiro": [
        "servidor Zabbix",
        "Servidor Linux"
      ]
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

```
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "manutenção": [],
      "hostid": "10160",
      "proxy_hostid": "0",
      "host": "servidor Zabbix",
      "estado": "0",
      "lastaccess": "0",
      "ipmi_authtype": "-1",
      "ipmi_privilege": "2",
      "ipmi_username": "",
      "ipmi_password": "",
      "ipmi_disable_until": "0",
      "snmp_disable_until": "0",
      "manutençãoid": "0",
      "maintenance_status": "0",
      "maintenance_type": "0",
      "maintenance_from": "0",
      "nome": "servidor Zabbix",
      "description": "0 servidor de monitoramento Zabbix.",
      "tls_connect": "1",
      "tls_accept": "1",
      "tls_issuer": "",
      "tls_subject": ""
    },
    {
      "manutenção": [],
      "hostid": "10167",
      "proxy_hostid": "0",
      "host": "Servidor Linux",
      "estado": "0",
      "lastaccess": "0",
      "ipmi_authtype": "-1",
      "ipmi_privilege": "2",
      "ipmi_username": "",
      "ipmi_password": "",
      "ipmi_disable_until": "0",
      "snmp_disable_until": "0",
      "manutençãoid": "0",
      "maintenance_status": "0",
      "maintenance_type": "0",
      "maintenance_from": "0",
      "nome": "Servidor Linux",
      "Descrição": "",
      "tls_connect": "1",
      "tls_accept": "1",
      "tls_issuer": "",
      "tls_subject": ""
    }
  ],
  "id": 1
}
```

Recuperando grupos de hosts

Recupere os nomes dos grupos que o host "Zabbix server" é membro, mas não detalhes do host em si.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["hostid"],
    "selectGroups": "estender",
    "filtro": {
      "hospedeiro": [
        "servidor Zabbix"
      ]
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 2
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10085",
      "grupos": [
        {
          "groupid": "2",
          "name": "Servidores Linux",
          "interno": "0",
          "bandeiras": "0"
        },
        {
          "groupid": "4",
          "name": "servidores Zabbix",
          "interno": "0",
          "bandeiras": "0"
        }
      ]
    }
  ],
  "id": 2
}
```

Recuperando modelos vinculados

Recupere os IDs e os nomes dos modelos vinculados ao host "10084".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["hostid"],
    "selectParentTemplates": [
      "modelo",
      "nome"
    ],
    "hostids": "10084"
  },
  "id": 1,
  "auth": "70785d2b494a7302309b48afcdb3a401"
}
```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10084",
      "parentModelos": [
        {
          "nome": "Linux",
          "templateid": "10001"
        },
        {
          "nome": "Servidor Zabbix",
          "templateid": "10047"
        }
      ]
    }
  ],
  "id": 1
}

```

Retrieving hosts by template

Retrieve hosts that have the "10001" (Linux by Zabbix agent) template linked to them.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "output": ["hostid", "name"],
    "templateids": "10001"
  },
  "auth": "70785d2b494a7302309b48afcdb3a401",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "templateid": "10001",
      "hosts": [
        {
          "hostid": "10084",
          "name": "Zabbix server"
        },
        {
          "hostid": "10603",
          "name": "Host 1"
        },
        {
          "hostid": "10604",
          "name": "Host 2"
        }
      ]
    }
  ],
  "id": 1
}

```

Pesquisando por dados de inventário do host

Recupere hosts que contenham "Linux" no campo "OS" do inventário de hosts.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": [
      "hospedeiro"
    ],
    "selectInventário": [
      "os"
    ],
    "pesquisarInventário": {
      "os": "Linux"
    }
  },
  "id": 2,
  "auth": "7f9e00124c75e8f25facd5c093f3e9a0"
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10084",
      "host": "servidor Zabbix",
      "inventário": {
        "os": "Linux Ubuntu"
      }
    },
    {
      "hostid": "10107",
      "host": "Servidor Linux",
      "inventário": {
        "os": "Linux Mint"
      }
    }
  ],
  "id": 1
}
```

Pesquisando por tags de host

Recupere os hosts que possuem a tag "Nome do host" igual a "Servidor Linux".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["hostid"],
    "selectTags": "estender",
    "evaltype": 0,
    "Tag": [
      {
        "tag": "Nome do host",
        "value": "Servidor Linux",
        "operador": 1
      }
    ]
  },
  "auth": "7f9e00124c75e8f25facd5c093f3e9a0",
  "id": 1
}
```

```
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10085",
      "Tag": [
        {
          "tag": "Nome do host",
          "value": "Servidor Linux"
        },
        {
          "tag": "SO",
          "valor": "RHEL 7"
        }
      ]
    }
  ],
  "id": 1
}
```

Recupere hosts que tenham essas tags não apenas em nível de host, mas também em seus modelos pai vinculados.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["nome"],
    "tags": [{"tag": "A", "value": "1", "operator": "0"}],
    "herdadoTags": verdadeiro
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10623",
      "name": "Sala de PC 1"
    },
    {
      "hostid": "10601",
      "nome": "Escritório"
    }
  ],
  "id": 1
}
```

Pesquisando host com tags e tags de modelo

Recuperar um host com tags e todas as tags vinculadas ao pai modelos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
```

```

    "saída": ["nome"],
    "hostids": 10502,
    "selectTags": ["tag", "valor"],
    "selectInheritedTags": ["tag", "valor"]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10502",
      "nome": "Área de trabalho",
      "Tag": [
        {
          "marca": "A",
          "valor": "1"
        }
      ],
      "herdadoTags": [
        {
          "marca": "B",
          "valor": "2"
        }
      ]
    }
  ],
  "id": 1
}

```

Pesquisando hosts por gravidade do problema

Recupere hosts que tenham problemas de "Desastre".

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["nome"],
    "gravidades": 5
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10160",
      "name": "servidor Zabbix"
    }
  ],
  "id": 1
}

```

Recupere hosts que tenham problemas "Average" e "High".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.get",
  "parâmetros": {
    "saída": ["nome"],
    "gravidades": [3, 4]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "20170",
      "nome": "Banco de dados"
    },
    {
      "hostid": "20183",
      "nome": "estação de trabalho"
    }
  ],
  "id": 1
}
```

Veja também

- [Grupo de hosts](#)
- [Modelo](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)

Fonte

CHost::get() em ui/include/classes/api/services/CHost.php.

host.massadd

Descrição

`object host.massadd(object parameters)`

Este método permite adicionar simultaneamente vários objetos relacionados a todos os hospedeiros dados.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo os IDs dos hosts a serem atualizados e o objetos para adicionar a todos os hosts.

O método aceita os seguintes parâmetros.

[Parâmetro]	Tipo	Descrição
<code>hosts</code>	(obrigatório) object/array Hosts	a serem atualizados. Os hosts devem ter a propriedade <code>hostid</code> definida.
<code>grupos</code>	objeto/array Grupos de hosts	para adicionar aos hosts fornecidos.
<code>interfaces</code>	object/array Interfaces de host	a serem criadas para os hosts fornecidos.
<code>macros</code>	object/array Macros do usuário	a serem criados para os hosts fornecidos.
<code>templates</code>	object/array Modelos	para vincular aos hosts fornecidos.

Os templates devem ter a propriedade `templateid` definida.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts atualizados sob a propriedade `hostids`.

Exemplos

Adicionando macros

Adicione duas novas macros a dois hosts.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.massadd",
  "parâmetros": {
    "anfitriões": [
      {
        "hostid": "10160"
      },
      {
        "hostid": "10167"
      }
    ],
    "macros": [
      {
        "macro": "${TEST1}",
        "valor": "MACROTEST1"
      },
      {
        "macro": "${TEST2}",
        "valor": "MACROTEST2",
        "description": "Descrição do teste"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10160",
      "10167"
    ]
  },
  "id": 1
}
```

Veja também

- [host.update](#)
- [Grupo de hosts](#)
- [Modelo](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)

Fonte

CHost::massAdd() em ui/include/classes/api/services/CHost.php.

host.massremove

Descrição

object host.massremove(object parameters)

Este método permite remover objetos relacionados de vários hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo os IDs dos hosts a serem atualizados e o objetos que devem ser removidos.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **hostids**
(obrigatório)|string/array|IDs dos hosts a serem atualizados.| [groupids|string/array|Grupos de hosts dos quais remover os hosts fornecidos.| [interfaces|object/array|Interfaces de host a serem removidas dos hosts fornecidos.

O objeto de interface de host deve ter as propriedades ip, dns e port definidas.| [macros|string/array|Macros de usuário a serem excluídos dos hosts fornecidos.| [templateids|string/array|Modelos para desvincular dos hosts fornecidos.| [templateids_clear|string/array|Modelos para desvincular e limpar dos hosts fornecidos.]

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts atualizados sob a propriedade `hostids`.

Exemplos

Desvinculando modelos

Desvincule um modelo de dois hosts e exclua todos os modelos entidades.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.massremove",
  "parâmetros": {
    "hostids": ["69665", "69666"],
    "templateids_clear": "325"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "69665",
      "69666"
    ]
  },
  "id": 1
}
```

Veja também

- [host.update](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)

Fonte

CHost::massRemove() em ui/include/classes/api/services/CHost.php.

host.massupdate

Descrição

object host.massupdate(object parameters)

Este método permite substituir ou remover simultaneamente objetos relacionados e atualizar propriedades em vários hosts.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo os IDs dos hosts a serem atualizados e o propriedades que devem ser atualizadas.

Além das **propriedades padrão do host**, o método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----| |**hosts**
(obrigatório)|object/array|**Hosts** a serem atualizados.

Os hosts devem ter o **hostid** propriedade definida. | [grupos|objeto/array|Host **grupos** para substituir os grupos de hosts atuais aos quais os hosts pertencem.

Os grupos de hosts devem ter a propriedade **groupid** definido. | [interfaces|object/array|Host **interfaces** para substituir as interfaces de host atuais nos hosts fornecidos. | [inventory|object|Host **inventory** propriedades.

O modo de inventário do host não pode ser atualizado usando o parâmetro **inventory**, use **inventory_mode**. | [macros|object/array|**Macros de usuário** para substituir as macros de usuário atuais nos hosts fornecidos. | [templates|object/array|**Templates** para substituir os templates atualmente vinculados nos hosts fornecidos.

Os templates devem ter a propriedade **templateid** definida. | [templates_clear|object/array|**Templates** para desvincular e limpar dos hosts fornecidos.

Os templates devem ter a propriedade **templateid** definida. |

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts atualizados sob a propriedade **hostids**.

Exemplos

Habilitando vários hosts

Habilite o monitoramento de dois hosts, ou seja, defina seu status como 0.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.massupdate",
  "parâmetros": {
    "anfitriões": [
      {
        "hostid": "69665"
      },
      {
        "hostid": "69666"
      }
    ],
    "estado": 0
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "69665",
      "69666"
    ]
  },
  "id": 1
}
```

Veja também

- [host.update](#)
- [host.massadd](#)

- [host.massremove](#)
- [Grupo de hosts](#)
- [Modelo](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)

Fonte

CHost::massUpdate() em ui/include/classes/api/services/CHost.php.

host.update

Descrição

`object host.update(object/array hosts)`

Este método permite atualizar os hosts existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do host a serem atualizadas.

A propriedade `hostid` deve ser definida para cada host, todos os outros propriedades são opcionais. Somente as propriedades fornecidas serão atualizadas, todas outros permanecerão inalterados.

Observe, no entanto, que atualizar o nome técnico do host também atualizará o nome visível do host (se não fornecido ou vazio) pelo técnico do host valor do nome.

Além das [propriedades padrão do host](#), o método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição| |-----|-----|-----|-----| |groups|object/array|Host [groups](#) para substituir os grupos de hosts atuais aos quais o host pertence.

Os grupos de hosts devem ter o `groupid` propriedade definida. Todos os grupos de hosts que não estão listados na solicitação serão desvinculados.| |interfaces|object/array|Host [interfaces](#) para substituir as interfaces de host atuais.

Todas as interfaces que não estiverem listadas na solicitação serão removidas.| |tags|object/array|Host [tags](#) para substituir as tags de host atuais.

Todas as tags que não estiverem listadas na solicitação serão removidas. | |inventário|objeto|Host [inventário](#) propriedades.| |macros|object/array|[Macros de usuário](#) para substituir as macros de usuário atuais.

Todas as macros que não estiverem listadas na solicitação serão removidas.| |templates|object/array|[Templates](#) para substituir os templates atualmente vinculados. Todos os modelos que não estiverem listados na solicitação serão apenas desvinculados.

Os modelos devem ter a propriedade `templateid` definida.| |templates_clear|object/array|[Templates](#) para desvincular e limpar do host.

Os templates devem ter a propriedade `templateid` definida.]

Ao contrário do frontend Zabbix, quando `name` (visível nome do host) é o mesmo que `host` (nome técnico do host), atualizando o host via API não atualizará automaticamente `name`. Ambas as propriedades precisam ser atualizado explicitamente.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos hosts atualizados sob a propriedade `hostids`.

Exemplos

Habilitando um host

Habilite o monitoramento do host, ou seja, defina seu status como 0.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.update",
  "parâmetros": {
    "hostid": "10126",
    "estado": 0
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
```



```
    "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10126"
    ]
  },
  "id": 1
}
```

Desvinculando modelos

Desvincule e limpe dois modelos do host.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.update",
  "parâmetros": {
    "hostid": "10126",
    "templates_clear": [
      {
        "templateid": "10124"
      },
      {
        "templateid": "10125"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10126"
    ]
  },
  "id": 1
}
```

Atualizando macros de host

Substitua todas as macros de host por duas novas.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "host.update",
  "parâmetros": {
    "hostid": "10126",
    "macros": [
      {
        "macro": "{$PASS}",
        "valor": "senha"
      },
      {

```

```

        "macro": "${DISC}",
        "valor": "sda",
        "description": "Descrição atualizada"
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10126"
    ]
  },
  "id": 1
}

```

Atualizando o inventário do host

Alterar o modo de inventário e adicionar local

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "host.update",
  "parâmetros": {
    "hostid": "10387",
    "modo_inventário": 0,
    "inventário": {
      "location": "Letônia, Riga"
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10387"
    ]
  },
  "id": 1
}

```

Atualizando tags de host

Substitua todas as tags de host por uma nova.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "host.update",
  "parâmetros": {
    "hostid": "10387",
    "Tag": {
      "tag": "S0",
      "valor": "CentOS 7"
    }
  }
}

```

```

    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10387"
    ]
  },
  "id": 1
}

```

Updating host encryption

Update the host "10590" to use PSK encryption only for connections from host to Zabbix server, and change the PSK identity and PSK key. Note that the host has to be **pre-configured to use PSK**.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "host.update",
  "params": {
    "hostid": "10590",
    "tls_connect": 1,
    "tls_accept": 2,
    "tls_psk_identity": "PSK 002",
    "tls_psk": "e560cb0d918d26d31b4f642181f5f570ad89a390931102e5391d08327ba434e9"
  },
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "hostids": [
      "10590"
    ]
  },
  "id": 1
}

```

Veja também

- [host.massadd](#)
- [host.massupdate](#)
- [host.massremove](#)
- [Grupo de hosts](#)
- [Modelo](#)
- [Do utilizador macro](#)
- [Hospedeiro interface](#)
- [Inventário do host](#)
- [Tag do host](#)

Fonte

CHost::update() em ui/include/classes/api/services/CHost.php.

Host descoberto

Esta classe foi projetada para funcionar com hosts descobertos.

Referências de objetos:

- [Descoberto host](#)

Métodos disponíveis:

- [dhost.get](#) - recuperar descoberto anfitriões

> Objeto host descoberto

Os objetos a seguir estão diretamente relacionados à API `dhost`.

Host descoberto

Note:

Host descoberto é criado pelo servidor Zabbix e não pode ser modificado por meio da API.

O objeto `host descoberto` contém informações sobre um `host descoberto` por uma regra de descoberta de rede. Tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |dhostid|string|ID do host descoberto.|
[druleid|string|ID da regra de descoberta que detectou o host.| |lastdown|timestamp|Hora em que o host descoberto foi desativado pela última vez.| |lastup|timestamp|Horário em que o host descoberto foi ativado pela última vez.| |status|inteiro|Se o host descoberto está ativo ou inativo. Um host está ativo se tiver pelo menos um serviço descoberto ativo.

Valores possíveis:
0 - host ativo;
1 - host desativado.|

```
#####dhost.get {#manual-api-reference-dhost-get}
```

Descrição

`integer/array dhost.get(object parameters)`

O método permite recuperar hosts descobertos de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----| |dhostids|string/array|Retorna apenas hosts descobertos com os IDs fornecidos.| |druleids|string/array|Retorna apenas hosts descobertos que foram criados pelas regras de descoberta fornecidas.| |dserviceids|string/array|Retorna apenas os hosts descobertos que estão executando os serviços fornecidos.| |selectDRules|query|Retorne uma propriedade **drules** com uma matriz das regras de descoberta que detectaram o host.| |selectDServices|query|Retorne uma propriedade **dservices** com os serviços descobertos em execução no host.

Suporta count.| |limitSelects|integer|Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:
selectDServices - os resultados serão classificados por dserviceid.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: dhostid e druleid.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no [comentário de referência](#).| |editável|boolean|^| |excluiPesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro `countOutput` tiver sido usado.

Exemplos

Recuperar hosts descobertos por regra de descoberta

Recupere todos os hosts e os serviços descobertos que estão executando foram detectados pela regra de descoberta "4".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dhost.get",
  "parâmetros": {
    "saída": "estender",
    "selectDServices": "estender",
    "druleidas": "4"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "dserviços": [
        {
          "dserviceid": "1",
          "dhostid": "1",
          "tipo": "4",
          "chave_": "",
          "valor": "",
          "porta": "80",
          "estado": "0",
          "lastup": "1337697227",
          "lastdown": "0",
          "dcheckid": "5",
          "ip": "192.168.1.1",
          "dns": "estação.empresa.lan"
        }
      ],
      "dhostid": "1",
      "druleid": "4",
      "estado": "0",
      "lastup": "1337697227",
      "lastdown": "0"
    },
    {
      "dserviços": [
        {
          "dserviceid": "2",
          "dhostid": "2",
          "tipo": "4",
          "chave_": "",
          "valor": "",
          "porta": "80",
          "estado": "0",
          "lastup": "1337697234",
          "lastdown": "0",
          "dcheckid": "5",
          "ip": "192.168.1.4",
          "dns": "john.company.lan"
        }
      ],
      "dhostid": "2",
      "druleid": "4",
      "estado": "0",
      "lastup": "1337697234",

```

```

        "lastdown": "0"
    },
    {
        "dserviços": [
            {
                "dserviceid": "3",
                "dhostid": "3",
                "tipo": "4",
                "chave_": "",
                "valor": "",
                "porta": "80",
                "estado": "0",
                "lastup": "1337697234",
                "lastdown": "0",
                "dcheckid": "5",
                "ip": "192.168.1.26",
                "dns": "impressora.empresa.lan"
            }
        ],
        "dhostid": "3",
        "druleid": "4",
        "estado": "0",
        "lastup": "1337697234",
        "lastdown": "0"
    },
    {
        "dserviços": [
            {
                "dserviceid": "4",
                "dhostid": "4",
                "tipo": "4",
                "chave_": "",
                "valor": "",
                "porta": "80",
                "estado": "0",
                "lastup": "1337697234",
                "lastdown": "0",
                "dcheckid": "5",
                "ip": "192.168.1.7",
                "dns": "mail.company.lan"
            }
        ],
        "dhostid": "4",
        "druleid": "4",
        "estado": "0",
        "lastup": "1337697234",
        "lastdown": "0"
    }
],
    "id": 1
}

```

Veja também

- [Descoberto serviço](#)
- [Regra de descoberta](#)

Fonte

CDHost::get() em ui/include/classes/api/services/CDHost.php.

Imagem

Esta classe é projetada para trabalhar com imagens.

Referências de objetos:

- **Imagem**

Métodos disponíveis:

- **image.create** - criar novo imagens
- **image.delete** - excluir imagens
- **image.get** - recupera imagens
- **image.update** - atualizar imagens

> Objeto de imagem

Os objetos a seguir estão diretamente relacionados à API **image**.

Imagem

O objeto de imagem tem as seguintes propriedades.

Propriedade	Tipo	Descrição
imageid	string	(somente leitura) ID da imagem.
name (requerido)	string	Nome da imagem.
imagetype	integer	Tipo de imagem. Valores possíveis: 1 - (padrão) ícone; 2 - imagem de fundo.

image.create

Descrição

`object image.create(object/array images)`

Este método permite criar novas imagens.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver **Do utilizador funções** Para maiores informações.

Parâmetros

(objeto/array) Imagens para criar.

Além das **propriedades de imagem padrão**, o O método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| **nome**
(obrigatório)|string|Nome da imagem.| **tipo de imagem**
(obrigatório)|inteiro|Tipo de imagem.

Valores possíveis:
1 - ícone (padrão);
2 - imagem de fundo.| **imagem**
(obrigatório)|string|imagem codificada em Base64. O tamanho máximo da imagem codificada é 1 MB. O tamanho máximo pode ser ajustado alterando o valor da constante ZBX_MAX_IMAGE_SIZE.
Os formatos de imagem suportados são: PNG, JPEG, GIF.|

Valores de retorno

(object) Retorna um objeto contendo os IDs das imagens criadas sob a propriedade **imageids**. A ordem dos IDs retornados corresponde à ordem das imagens passadas.

Exemplos

Crie uma imagem

Crie um ícone de nuvem.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "imagem.criar",
  "parâmetros": {
    "tipo de imagem": 1,
    "name": "Cloud_(24)",
    "Imagem": "iVBORwOKGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAABHNCSVQICAgIfAhkiAAAAAlwSFlzAAACmAAAApg
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "imageids": [
      "188"
    ]
  },
  "id": 1
}
```

Fonte

CIImage::create() em ui/include/classes/api/services/CIImage.php.

image.delete

Descrição

object image.delete(array imageIds)

Este método permite excluir imagens.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(array) IDs das imagens a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das imagens excluídas sob a propriedade imageids.

Exemplos

Excluir várias imagens

Apague duas imagens.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "imagem.delete",
  "parâmetros": [
    "188",
    "192"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:


```
{
  "jsonrpc": "2.0",
  "resultado": {
    "imageids": [
      "188",
      "192"
    ]
  },
  "id": 1
}
```

Fonte

CImage::delete() em ui/include/classes/api/services/CImage.php.

image.get

Descrição

integer/array image.get(object parameters)

O método permite recuperar imagens de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

Parâmetro	Tipo	Descrição
imageids	string/array	Retorna apenas imagens com os IDs fornecidos.
sysmapids	string/array	Retorna imagens que são usadas nos mapas fornecidos.
select_image	flag	Retorne uma propriedade image com a imagem codificada em Base64.
sortfield	string/array	Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: imageid e name.

Estes parâmetros sendo comuns para todos os métodos get são descritos em detalhes no [comentário de referência](#).

Parâmetro	Tipo	Descrição
countOutput	boolean	Estes parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência.
editável	boolean	
excludePesquisa	boolean	
filtro	objeto	
limite	inteiro	
saída	consulta	
preservekeys	boolean	
pesquisar	objeto	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
ordem de classificação	cadeia/matriz	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar uma imagem

Recupere todos os dados da imagem com ID "2".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "imagem.get",
  "parâmetros": {
    "saída": "estender",
    "select_image": verdadeiro,
    "imageids": "2"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "imagemid": "2",
      "tipo de imagem": "1",
      "name": "Cloud_(24)",
      "Imagem": "iVBORwOKGgoAAAANSUhEUgAAABgAAAANCAYAAACzbK7QAAAAAHNCSVQICAgIfAhkiAAAAA1wSFlzAAACmAA
    }
  ],
  "id": 1
}
```

Fonte

CImage::get() em ui/include/classes/api/services/CImage.php.

imagem.atualização

Descrição

`object image.update(object/array images)`

Este método permite atualizar as imagens existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades da imagem a serem atualizadas.

A propriedade `imageid` deve ser definida para cada imagem, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas os outros permanecerão inalterados.

Além das [propriedades de imagem padrão](#), o O método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| |image|string|imagem codificada em Base64. O tamanho máximo da imagem codificada é 1 MB. O tamanho máximo pode ser ajustado alterando o valor da constante ZBX_MAX_IMAGE_SIZE.
Os formatos de imagem suportados são: PNG, JPEG, GIF.]

Valores de retorno

(object) Retorna um objeto contendo os IDs das imagens atualizadas sob a propriedade `imageids`.

Exemplos

Renomear imagem

Renomeie a imagem para "Ícone de nuvem".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "imagem.atualização",
  "parâmetros": {
    "imagemid": "2",
    "name": "Ícone de nuvem"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
```

```

        "imageids": [
            "2"
        ]
    },
    "id": 1
}

```

Fonte

CImage::update() em ui/include/classes/api/services/CImage.php.

informações da API

Essa classe foi projetada para recuperar metainformações sobre a API.

Métodos disponíveis:

- **apiinfo.version** - recuperando a versão da API do Zabbix

apiinfo.versão

Descrição

string apiinfo.version(array)

Este método permite recuperar a versão da API do Zabbix.

::: não importante Este método está disponível apenas para não autenticados usuários e deve ser chamado sem o parâmetro auth no JSON-RPC solicitação. :::

Parâmetros

(array) O método aceita um array vazio.

Valores de retorno

(string) Retorna a versão da API do Zabbix.

A partir do Zabbix 2.0.4 a versão da API corresponde à versão do Zabbix.

Exemplos

Recuperando a versão da API

Recupera a versão da API do Zabbix.

Requisição:

```

{
    "jsonrpc": "2.0",
    "method": "apiinfo.version",
    "params": [],
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": "4.0.0",
    "id": 1
}

```

Fonte

CAPIInfo::version() em ui/include/classes/api/services/CAPIInfo.php.

####Interface do host {#manual-api-reference-hostinterface}

Esta classe foi projetada para trabalhar com interfaces de host.

Referências de objetos:

- **Hospedeiro interface**

Métodos disponíveis:

- **hostinterface.create** - criando novas interfaces de host
- **hostinterface.delete** - excluindo interfaces de host
- **hostinterface.get** - recuperando interfaces de host
- **hostinterface.massadd** - adicionando interfaces de host a hosts
- **hostinterface.massremove** - removendo interfaces de host de hosts
- **hostinterface.replacehostinterfaces** - substituindo interfaces de host em um host
- **hostinterface.update** - atualizando interfaces de host

> Objeto de interface do host

Os objetos a seguir estão diretamente relacionados à API `hostinterface`.

Interface do host

O objeto de interface do host tem as seguintes propriedades.

::: não importante Observe que IP e DNS são necessários. Se você fizer não quiser usar o DNS, defina-o como uma string vazia. :::

[Propriedade|**Tipo**|Descrição] |-----|-----|-----|-----| [disponível|inteiro|(somente leitura) Disponibilidade da interface do host.

Os valores possíveis são:
0 - (padrão) desconhecido;
1 - disponível;
2 - indisponível.] |details|array|Objeto adicional para interface. **Obrigatório** se o 'tipo' da interface for SNMP.|disable_until|timestamp|(readonly) O próximo horário de polling de uma interface de host indisponível.|dns
(obrigatório)|string|nome DNS usado pela interface.

Pode estar vazio se a conexão for feita via IP.|error|string|(readonly) Texto de erro se a interface do host não estiver disponível.|errors_from|timestamp|(readonly) Hora em que a interface do host ficou indisponível.|**hostid**
(obrigatório)|string|ID do host ao qual a interface pertence.|interfaceid|string|(readonly) ID da interface.|ip
(obrigatório)|string|endereço IP usado pela interface.

Pode estar vazio se a conexão for feita via DNS.|main
(obrigatório)|integer|Se a interface é usada como padrão no host. Apenas uma interface de algum tipo pode ser definida como padrão em um host.

Os valores possíveis são:
0 - não padrão;
1 - padrão.|port
(obrigatório)|string|Número da porta usada pela interface. Pode conter macros de usuário.|tipo
(obrigatório)|inteiro|Tipo de interface.

Os valores possíveis são:
1 - agente;
2 - SNMP;
3 - IPMI;
4 - JMX.
|useip
(obrigatório)|inteiro|Se a conexão deve ser feita via IP.

Os valores possíveis são:
0 - conectar usando o nome DNS do host;
1 - conecte-se usando o endereço IP do host para esta interface do host.]

Detalhes de Tag

O objeto de detalhes tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----|-----| **versão**
(obrigatório)|inteiro|versão da interface SNMP.

Os valores possíveis são:
1 - SNMPv1;
2 - SNMPv2c;
3 - SNMPv3|bulk|inteiro|Se usar solicitações SNMP em massa.

Os valores possíveis são:
0 - não usar solicitações em massa;
1 - (padrão) - usar solicitações em massa.|community|string|Comunidade SNMP (obrigatório). Usado apenas por interfaces SNMPv1 e SNMPv2.|securityname|string|nome de segurança SNMPv3. Usado apenas por interfaces SNMPv3.|nível de segurança|inteiro|nível de segurança SNMPv3. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - noAuthNoPriv;
1 - authNoPriv;
2 - authPriv.|authpassphrase|string|Senha de autenticação SNMPv3. Usado apenas por interfaces SNMPv3.|privpassphrase|string|Senha de privacidade SNMPv3. Usado apenas por interfaces SNMPv3.|authprotocol|inteiro|protocolo de autenticação SNMPv3. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - MD5;
1 - SHA1;
2 - SHA224;
3 - SHA256;
4 - SHA384;
5 - SHA512.|privprotocol|integer|SNMPv3 protocolo de privacidade. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - DES;
1 - AES128;
2 - AES192;
3 - AES256;
4 - AES192C;
5 - AES256C.|contextname|string|SNMPv3 nome do contexto. Usado apenas por interfaces SNMPv3.]

`hostinterface.create`

Descrição

```
object hostinterface.create(object/array hostInterfaces)
```

Este método permite criar novas interfaces de host.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(objeto/array) Interfaces de host para criar. O método aceita host interfaces com a [interface de host padrão propriedades](#).

Valores de retorno

(object) Retorna um objeto contendo os IDs do host criado interfaces sob a propriedade `interfaceids`. A ordem do retorno Os IDs correspondem à ordem das interfaces de host passadas.

Exemplos

Crie uma nova interface

Crie uma interface de agente IP secundária no host "30052".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostinterface.create",
  "parâmetros": {
    "hostid": "30052",
    "principal": "0",
    "tipo 1",
    "useip": "1"
    "ip": "127.0.0.1",
    "dns": "",
    "porta": "10050",
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "interfaceids": [
      "30062"
    ]
  },
  "id": 1
}
```

Crie uma interface com detalhes do SNMP

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostinterface.create",
  "parâmetros": {
    "hostid": "10456",
    "principal": "0",
    "tipo 2",
    "useip": "1",
    "ip": "127.0.0.1",
    "dns": "",
    "porta": "1601",
    "detalhes": {
      "versão 2",
      "em massa": "1",
      "community": "{$SNMP_COMMUNITY}"
    }
  }
}
```

```

    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "interfaceids": [
      "30063"
    ]
  },
  "id": 1
}

```

Veja também

- [hostinterface.massadd](#)
- [host.massadd](#)

Fonte

CHostInterface::create() em ui/include/classes/api/services/CHostInterface.php.

hostinterface.delete

Descrição

object hostinterface.delete(array hostInterfaceIds)

Este método permite excluir interfaces de host.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs das interfaces de host a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host excluído interfaces sob a propriedade interfaceids.

Exemplos

Excluir uma interface de host

Exclua a interface do host com ID 30062.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "método": "hostinterface.delete",
  "parâmetros": [
    "30062"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {

```

```

        "interfaceids": [
            "30062"
        ]
    },
    "id": 1
}

```

Veja também

- [hostinterface.massremove](#)
- [host.massremove](#)

Fonte

CHostInterface::delete() em ui/include/classes/api/services/CHostInterface.php.

hostinterface.get

Descrição

integer/array hostinterface.get(object parameters)

O método permite recuperar interfaces de host de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
hostids	string/array	Retorna apenas interfaces de host usadas pelos hosts fornecidos.
interfaceids	string/array	Retorna apenas interfaces de host com os IDs fornecidos.
itemids	string/array	Retorna apenas interfaces de host usadas pelos itens fornecidos.
triggerids	string/array	Retorna apenas interfaces de host usadas pelos itens nos triggers fornecidos.
selectItems	query	Retorna uma propriedade items com os itens que usam a interface.
selectHosts	query	Suporta count. Retorna uma propriedade hosts com um array de hosts que usam a interface.
limitSelects	integer	Limita o número de registros retornados por subseleções.
sortfield	string/array	Aplica-se às seguintes subseleções: selectItems . Ordena o resultado pelas propriedades fornecidas.
countOutput	boolean	Possible values are: interfaceid , dns , ip . Esses parâmetros, sendo comuns para todos os métodos get são descritos detalhadamente na página de reference commentary page.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	

Parâmetro	Tipo	Descrição
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar interfaces do host

Recupere todos os dados sobre as interfaces usadas pelo host "30057."

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "hostinterface.get",
  "params": {
    "output": "extend",
    "hostids": "30057"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "interfaceid": "50039",
      "hostid": "30057",
      "main": "1",
      "type": "1",
      "useip": "1",
      "ip": "::1",
      "dns": "",
      "port": "10050",
      "available": "0",
      "error": "",
      "errors_from": "0",
      "disable_until": "0",
      "details": []
    },
    {
      "interfaceid": "55082",
      "hostid": "30057",
      "main": "0",
      "type": "1",
      "useip": "1",
      "ip": "127.0.0.1",
      "dns": "",
      "port": "10051",
      "available": "0",
      "error": "",
      "errors_from": "0",
      "disable_until": "0",
      "details": {
        "version": "2",
        "bulk": "0",
        "community": "${SNMP_COMMUNITY}"
      }
    }
  ]
}
```



```

    }
    }
],
    "id": 1
}

```

Veja também

- [Host](#)
- [Item](#)

Fonte

CHostInterface::get() em ui/include/classes/api/services/CHostInterface.php.

hostinterface.massadd

Descrição

`object hostinterface.massadd(object parameters)`

Este método permite adicionar simultaneamente interfaces de host a vários anfitriões.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros contendo as interfaces do host a serem criadas os hospedeiros dados.

O método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| **hosts**
(obrigatório)|object/array|Hosts a serem atualizados.

Os hosts devem ter a propriedade `hostid` definida.| **interfaces**
(obrigatório)|object/array|**Interfaces de host** para criar nos hosts fornecidos.]

Valores de retorno

(object) Retorna um objeto contendo os IDs do host criado interfaces sob a propriedade `interfaceids`.

Exemplos

Criando interfaces

Crie uma interface em dois hosts.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "hostinterface.massadd",
    "parâmetros": {
        "anfitriões": [
            {
                "hostid": "30050"
            },
            {
                "hostid": "30052"
            }
        ],
        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "principal": 0,
            "porta": "10050",
            "tipo": 1,
            "usei": 1
        }
    }
}

```

```

    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "interfaceids": [
      "30069",
      "30070"
    ]
  },
  "id": 1
}

```

Veja também

- [hostinterface.create](#)
- [host.massadd](#)
- [Host](#)

Fonte

CHostInterface::massAdd() em ui/include/classes/api/services/CHostInterface.php.

hostinterface.massremove

Descrição

object hostinterface.massremove(object parameters)

Este método permite remover interfaces de host dos hosts fornecidos.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros contendo os IDs dos hosts a serem atualizados e as interfaces a serem removidas.

[Parâmetro|Tipo|Descrição| |-----|-----|-----| |**hostids**
(obrigatório)|string/array|IDs dos hosts a serem atualizados.| **interfaces**
(obrigatório)|objeto/array|Interfaces de host a serem removidas dos hosts fornecidos.

O objeto de interface de host deve ter as propriedades de ip, dns e porta definidas]

Valores de retorno

(object) Retorna um objeto contendo os IDs do host excluído interfaces sob a propriedade interfaceids.

Exemplos

Removendo interfaces

Remova a interface SNMP "127.0.0.1" de dois hosts.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "method": "hostinterface.massremove",
  "parâmetros": {
    "hostids": [
      "30050",
      "30052"
    ],
  },
}

```

```

        "interfaces": {
            "dns": "",
            "ip": "127.0.0.1",
            "porta": "161"
        }
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "interfaceids": [
            "30069",
            "30070"
        ]
    },
    "id": 1
}

```

Veja também

- [hostinterface.delete](#)
- [host.massremove](#)

Fonte

CHostInterface::massRemove() em ui/include/classes/api/services/CHostInterface.php.

hostinterface.replacehostinterfaces

Descrição

`object hostinterface.replacehostinterfaces(object parameters)`

Este método permite substituir todas as interfaces de host em um determinado host.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

Parâmetros (object) contendo o ID do host a ser atualizado e as novas interfaces de host.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| |**hostid**
(obrigatório)|string|ID do host a ser atualizado.| |**interfaces**
(obrigatório)|object/array|**Interfaces de host** para substituir as interfaces de host atuais.|

Valores de retorno

(object) Retorna um objeto contendo os IDs do host criado interfaces sob a propriedade `interfaceids`.

Exemplos

Substituindo interfaces de host

Substitua todas as interfaces de host por uma única interface de agente.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "method": "hostinterface.replacehostinterfaces",
    "parâmetros": {
        "hostid": "30052",
        "interfaces": {
            "dns": "",

```

```

        "ip": "127.0.0.1",
        "principal": 1,
        "porta": "10050",
        "tipo": 1,
        "usei": 1
    }
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "interfaceids": [
            "30081"
        ]
    },
    "id": 1
}

```

Veja também

- [host.update](#)
- [host.massupdate](#)

Fonte

CHostInterface::replaceHostInterfaces() em ui/include/classes/api/services/CHostInterface.php.

#####hostinterface.update {#manual-api-reference-hostinterface-update}

Descrição

`object hostinterface.update(object/array hostInterfaces)`

Este método permite atualizar interfaces de host existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) **Propriedades da interface do host** para ser atualizado.

A propriedade `interfaceid` deve ser definida para cada interface do host, todas outras propriedades são opcionais. Apenas as propriedades dadas serão atualizado, todos os outros permanecerão inalterados.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host atualizado interfaces sob a propriedade `interfaceids`.

Exemplos

Alterando uma porta de interface do host

Altere a porta de uma interface de host.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "hostinterface.update",
    "parâmetros": {
        "interfaceid": "30048",
        "porta": "30050"
    }
}

```

```

    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "interfaceids": [
      "30048"
    ]
  },
  "id": 1
}

```

Fonte

CHostInterface::update() em ui/include/classes/api/services/CHostInterface.php.

Item

Esta classe foi projetada para trabalhar com itens.

Referências de objetos:

- [Item](#)

Métodos disponíveis:

- [item.create](#) - criando novo Itens
- [item.delete](#) - excluindo itens
- [item.get](#) - recuperando itens
- [item.update](#) - atualizando itens

> Objeto de item

Os objetos a seguir estão diretamente relacionados à API `item`.

Item

Note:

Os itens da Web não podem ser criados, atualizados ou excluídos diretamente por meio de a API do Zabbix.

O objeto `item` tem as seguintes propriedades.

Propriedade	Tipo	Descrição
itemid	string (readonly)	ID do item.
atraso	(obrigatório) string	Intervalo de atualização do item. Aceita segundos ou uma unidade de tempo com sufixo (30s,1m,2h,1d). Opcionalmente, um ou mais intervalos personalizados podem ser especificados como intervalos flexíveis ou agendamento . Vários intervalos são separados por um ponto e vírgula. As macros do usuário podem ser usadas. Uma única macro deve preencher todo o campo. Não há suporte para várias macros em um campo ou macros misturadas com texto. Intervalos flexíveis podem ser escritos como duas macros separadas por uma barra (por exemplo, {\$FLEX_INTERVAL}/{FLEX_PERIOD}). Opcional para Zabbix trapper, itens dependentes e para Zabbix agent (ativo) com chave <code>mqtt.get</code> . hostid
hostid	(obrigatório) string	ID do host ou modelo ao qual o item pertence. Para operações de atualização, este campo é readonly. interfaceid
interfaceid	(obrigatório) string	ID da interface do host do item. Não obrigatório para itens de modelo. Opcional para itens internos, agente ativo, trapper, calculado, dependente e monitor de banco de dados. chave
chave	(obrigatório) string	Chave do item. nome
nome	(obrigatório) string	Nome do item. tipo
tipo	(obrigatório) inteiro	Tipo do item. Valores possíveis: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - Verificação simples; 5 - Zabbix interno; 7 - Agente Zabbix (ativo); 9 - Item da Web; 10 - Verificação externa; 11 - Monitor de banco de dados; 12 - Agente IPMI; 13 - Agente SSH; 14 - Agente Telnet; 15 - Calculado; 16 - Agente JMX; 17 - Armadilha SNMP; 18 - Item dependente ; 19 - Agente HTTP; 20 - Agente SNMP; 21 - Script url
url	(obrigatório) string	string de URL, necessária apenas para o tipo de item de agente HTTP. Suporta macros de usuário, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}. valor_tipo
valor_tipo	(obrigatório) inteiro	Tipo de informação do item. Valores possíveis: 0 - número flutuante; 1

- caractere;
2 - log;
3 - numérico sem sinal;
4 - texto.| |allow_traps|inteiro|campo de item do agente HTTP. Permitir preencher valor como no tipo de item trapper também.

0 - (padrão) Não permitir aceitar dados de entrada.
1 - Permitir aceitar dados de entrada.| |auth_type|integer|Usado apenas por itens do agente SSH ou itens do agente HTTP.

Valores possíveis do método de autenticação do agente SSH:
0 - (padrão) senha;
1 - chave pública.

método de autenticação do agente HTTP valores possíveis:
0 - (padrão) nenhum
1 - básico
2 - NTLM
3 - Kerberos| |descricao|string|Descrição do item.| |error|string|(readonly) Texto de erro se houver problemas ao atualizar o item.| |flags|inteiro|(somente leitura) Origem do item.

Valores possíveis:
0 - um item simples;
4 - um item descoberto.| |follow_redirects|inteiro|campo de item do agente HTTP. Siga os redirecionamentos de resposta ao agrupar dados.

0 - Não siga os redirecionamentos.
1 - (padrão) Siga os redirecionamentos.| |headers|object|campo de item do agente HTTP. Objeto com cabeçalhos de solicitação HTTP(S), onde o nome do cabeçalho é usado como chave e o valor do cabeçalho como valor.

Exemplo:
{ "User-Agent": "Zabbix" }| |history|string|Uma unidade de tempo de quanto tempo os dados do histórico devem ser armazenados. Também aceita macro de usuário.

Padrão: 90d.| |http_proxy|string|campo de item do agente HTTP. Cadeia de conexão do proxy HTTP(S).| |inventory_link|inteiro|ID do campo de inventário do host que é preenchido pelo item.

Consulte a [página de inventário do host](#) para obter uma lista de campos de inventário de host suportados e seus IDs.

Padrão: 0.| |ipmi_sensor|string|sensor IPMI. Usado apenas por itens do IPMI.| |jmx_endpoint|string|String de conexão personalizada do agente JMX.

Valor padrão:
service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi| |lastclock|timestamp|(readonly) Hora em que o item foi atualizado pela última vez.

Por padrão, apenas os valores que se enquadram nas últimas 24 horas são exibidos. Você pode estender esse período alterando o valor do parâmetro Período máximo de exibição do histórico na seção de menu **Administração → Geral**.| |lastns|integer|(readonly) Nanossegundos quando o item foi atualizado pela última vez.

Por padrão, apenas os valores que se enquadram nas últimas 24 horas são exibidos. Você pode estender esse período alterando o valor do parâmetro Período máximo de exibição do histórico na seção de menu **Administração → Geral**.| |lastvalue|string|(readonly) Último valor do item.

Por padrão, apenas os valores das últimas 24 horas são exibidos. Você pode estender esse período de tempo alterando o value do parâmetro Período máximo de exibição do histórico na seção de menu **Administration → General**.| |logtimefmt|string|Formato da hora nas entradas de log. Usado apenas por itens de log.| |master_itemid|integer|ID do item mestre.

Recursão de até 3 itens dependentes e contagem máxima de itens dependentes igual a 29999 são permitidas.

Requerido por itens dependentes.| |output_format|inteiro|campo de item do agente HTTP. A resposta deve ser convertida em JSON.

0 - (padrão) Armazenar raw.
1 - Converter em JSON.| |params|string|Parâmetros adicionais dependendo do tipo do item:
- script executado para itens SSH e Telnet;
- consulta SQL para itens do monitor de banco de dados;
- fórmula para itens calculados;
- o script para o item de script.| |parameters|array|Parâmetros adicionais para itens de script. Matriz de objetos com propriedades 'nome' e 'valor', onde o nome deve ser exclusivo.| |senha|string|Senha para autenticação. Usado por itens de verificação simples, SSH, Telnet, monitor de banco de dados, JMX e agente HTTP.

Quando usado por JMX, o nome de usuário também deve ser especificado junto com a senha ou ambas as propriedades devem ser deixadas em branco.| |post_type|inteiro|campo de item do agente HTTP. Tipo de corpo de dados de postagem armazenado na propriedade de postagens.

0 - (padrão) Dados brutos.
2 - Dados JSON.
3 - Dados XML.| |posts|string|campo de item do agente HTTP. Dados do corpo da solicitação HTTP(S). Usado com post_type.| |prevvalue|string|(readonly) Valor anterior do item.

Por padrão, apenas os valores das últimas 24 horas são exibidos. Você pode estender esse período alterando o valor do parâmetro Período máximo de exibição do histórico na seção de menu **Administração → Geral**.| |privatekey|string|Nome do arquivo de chave privada.| |publickey|string|Nome do arquivo de chave pública.| |query_fields|array|campo de item do agente HTTP. Parâmetros de consulta. Matriz de objetos com pares 'chave': 'valor', onde valor pode ser uma string vazia.| |request_method|integer|campo de item do agente HTTP. Tipo de método de solicitação.

0 - (padrão) GET
1 - POST
2 - PUT
3 - HEAD| |recuperar_modulo|inteiro|campo de item do agente HTTP. Qual parte da resposta deve ser armazenada.

0 - (padrão) Corpo.
1 - Cabeçalhos.
2 - O corpo e os cabeçalhos serão armazenados.

Para request_method HEAD somente 1 é o valor permitido.| |snmp_oid|string|SNMP OID.| |ssl_cert_file|string|campo de item do agente HTTP. Caminho do arquivo de chave SSL pública.| |ssl_key_file|string|campo de item do agente HTTP. Caminho do arquivo de chave SSL privada.| |ssl_key_password|string|campo de item do agente HTTP. Senha para arquivo de chave SSL.| |estado|inteiro|(somente leitura) Estado do item.

Valores possíveis:
0 - (padrão) normal;
1 - não suportado.| |status|inteiro|Status do item.

Valores possíveis:
0 - (padrão) item habilitado;
1 - item desabilitado.| |status_codes|string|campo de item do agente HTTP. Intervalos de códigos de status HTTP necessários separados por vírgulas. Também suporta macros de usuário como parte da lista separada por vírgulas.

Exemplo: 200,200-{\$M},{ \$M},200-400| |templateid|string|(somente leitura) ID do item de modelo pai.

Dica: Use a propriedade hostid para especificar o modelo ao qual o item pertence.| |timeout|string|Tempo limite da solicitação de sondagem de dados do item. Usado para agente HTTP e itens de script. Suporta macros de usuário.

padrão: 3s
valor máximo: 60s| |trapper_hosts|string|Hosts permitidos. Usado por itens de trapper ou itens de agente HTTP.| |trends|string|Uma unidade de tempo de quanto tempo os dados de tendências devem ser armazenados. Também aceita macro de usuário.

Padrão: 365d.| |unidades|string|Unidades de valor.| |username|string|Username para autenticação. Usado por verificação simples, SSH, Telnet, monitor de banco de dados, itens de agente JMX e HTTP.

Requerido por itens SSH e Telnet.

Quando usado por JMX, a senha também deve ser especificada junto com o nome de usuário ou ambas as propriedades devem ser deixadas em branco.| |uuid|string|Identificador único universal, usado para vincular item importado a itens já existentes. Usado apenas para itens em modelos. Gerado automaticamente, se não fornecido.

Para operações de atualização, este campo é somente leitura.| |valuemapid|string|ID do mapa de valores associado.| |verify_host|inteiro|campo de item do agente HTTP. Validar o nome do host na URL está no campo Nome Comum ou em um campo Nome Alternativo da Entidade do certificado do host.

0 - (padrão) Não validar.
1 - Validar.| |verify_peer|integer|campo de item do agente HTTP. Validar é o certificado do host autêntico.

0 - (padrão) Não validar.
1 - Validar.|

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object/array) Itens para criar.

Além das **propriedades padrão do item**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
preprocessing	array	Opções preprocessing de itens.
tags	array	Item tags .

Valores de retorno

(object) Retorna um objeto contendo os IDs dos itens criados na propriedade `itemids`. A ordem dos IDs retornados corresponde à ordem dos itens passados.

Exemplos**Criando um item**

Crie um item de agente Zabbix numérico com 2 tags de item para monitorar o espaço livre em disco no host com o ID "30074".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "name": "Free disk space on /home/joe/",
    "key_": "vfs.fs.size[/home/joe/,free]",
    "hostid": "30074",
    "type": 0,
    "value_type": 3,
    "interfaceid": "30084",
    "tags": [
      {
        "tag": "Disc usage"
      },
      {
        "tag": "Equipment",
        "value": "Workstation"
      }
    ],
    "delay": "30s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "24758"
    ]
  },
  "id": 1
}
```

Criando um item de inventário do host

Crie um item de agente Zabbix para preencher o campo de inventário "SO" do host.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "name": "uname",
    "key_": "system.uname",
    "hostid": "30021",
    "type": 0,
    "interfaceid": "30007",
    "value_type": 1,
    "delay": "10s",
    "inventory_link": 5
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "24759"
    ]
  },
  "id": 1
}
```

Criando um item com pré-processamento

Criar um item usando o multiplicador personalizado.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "name": "Device uptime",
    "key_": "sysUpTime",
    "hostid": "11312",
    "type": 4,
    "snmp_oid": "SNMPv2-MIB::sysUpTime.0",
    "value_type": 1,
    "delay": "60s",
    "units": "uptime",
    "interfaceid": "1156",
    "preprocessing": [
      {
        "type": 1,
        "params": "0.01",
        "error_handler": 1,
        "error_handler_params": ""
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
```

```

    "result": {
      "itemids": [
        "44210"
      ]
    },
    "id": 1
  }
}

```

Criando item dependente

Crie um item dependente para o item mestre com ID 24759. Somente dependências no mesmo host são permitidas, portanto, o mestre e o item dependente devem ter o mesmo hostid

Request:

```

{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "hostid": "30074",
    "name": "Dependent test item",
    "key_": "dependent.item",
    "type": 18,
    "master_itemid": "24759",
    "value_type": 2
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "44211"
    ]
  },
  "id": 1
}

```

Criar item de agente HTTP

Criar item de método de solicitação POST com pré-processamento de resposta JSON.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "url": "http://127.0.0.1/http.php",
    "query_fields": [
      {
        "mode": "json"
      },
      {
        "min": "10"
      },
      {
        "max": "100"
      }
    ],
    "interfaceid": "1",
    "type": 19,
    "hostid": "10254",
  }
}

```

```

    "delay": "5s",
    "key_": "json",
    "name": "HTTP agent example JSON",
    "value_type": 0,
    "output_format": 1,
    "preprocessing": [
      {
        "type": 12,
        "params": "$.random",
        "error_handler": 0,
        "error_handler_params": ""
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 2
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "23865"
    ]
  },
  "id": 3
}

```

Criar item de script

Crie uma coleta de dados simples usando um item de script.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "item.create",
  "params": {
    "name": "Script example",
    "key_": "custom.script.item",
    "hostid": "12345",
    "type": 21,
    "value_type": 4,
    "params": "var request = new CurlHttpRequest();\nreturn request.Post(\"https://postman-echo.com/post\");",
    "parameters": [
      {
        "name": "host",
        "value": "{HOST.CONN}"
      }
    ]
  },
  "timeout": "6s",
  "delay": "30s"
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 2
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "23865"
    ]
  }
}

```

```

    ],
    },
    "id": 3
}

```

Fonte

CItem::create() in ui/include/classes/api/services/CItem.php.

item.delete

Descrição

objeto item.delete(array itemIds)

Este método permite excluir itens.

Note:

Os itens da Web não podem ser excluídos pela API do Zabbix.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos itens a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos itens excluídos sob a propriedade itemids.

Exemplos

Excluindo vários itens

Exclua dois itens.

Itens dependentes e protótipos de itens são removidos automaticamente se o item mestre for excluído.

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "item.delete",
  "params": [
    "22982",
    "22986"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "22982",
      "22986"
    ]
  },
  "id": 1
}

```

Fonte

CItem::delete() em ui/include/classes/api/services/CItem.php.

item.get

Descrição

integer/array item.get(object parameters)

O método permite recuperar itens de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [itemids|string/array|Retorna apenas itens com os IDs fornecidos.] [groupids|string/array|Retorna apenas os itens que pertencem aos hosts dos grupos fornecidos.] [templateids|string/array|Retorna apenas os itens que pertencem aos modelos fornecidos.] [hostids|string/array|Retorna apenas os itens que pertencem aos hosts fornecidos.] [proxyids|string/array|Retorna apenas os itens que são monitorados pelos proxies fornecidos.] [interfaceids|string/array|Retorna apenas itens que usam as interfaces de host especificadas.] [graphids|string/array|Retorna apenas os itens que são usados nos gráficos fornecidos.] [triggerids|string/array|Retorna apenas os itens usados nos gatilhos fornecidos.] [webitems|flag|Incluir itens da web no resultado.] [inherited|boolean|Se definido como true, retornará apenas itens herdados de um modelo.] [templated|boolean|Se definido como true, retornará apenas os itens que pertencem aos modelos.] [monitored|boolean|Se definido como true, retornará apenas os itens habilitados que pertencem aos hosts monitorados.] [group|string|Retorna apenas os itens que pertencem a um grupo com o nome fornecido.] [host|string|Retorna apenas os itens que pertencem a um host com o nome fornecido.] [evaltype|integer|Regras para pesquisa de tags.

Valores possíveis:
0 - (padrão) E/Ou;
2 - Ou.
1 - Igual;
3 - Diferente;
4 - Não igual;
5 - Existe;
6 - Não existe.] [with_triggers|boolean|Se definido como true, retorne apenas os itens que são usados em gatilhos.] [selectHosts|query|Retorne uma propriedade **hosts** com uma matriz de hosts aos quais o item pertence.] [selectInterfaces|query|Retorne uma propriedade de **interfaces** com uma matriz de interfaces de host usadas pelo item.] [selectTriggers|query|Retorne uma propriedade **triggers** com os gatilhos nos quais o item é usado.

Suporta count.] [selectGraphs|query|Retorne uma propriedade **graphs** com os gráficos que contêm o item.

Suporta count.] [selectDiscoveryRule|query|Retorne uma propriedade **discoveryRule** com a regra LLD que criou o item.] [selectItemDiscovery|query|Retorne uma propriedade **itemDiscovery** com o objeto de descoberta de item. O objeto de descoberta de item vincula o item a um protótipo de item a partir do qual ele foi criado.

Ele tem as seguintes propriedades:
itemdiscoveryid - (string) ID da descoberta de item;
itemid - (string) ID do item descoberto;
parent_itemid - (string) ID do protótipo do item a partir do qual o item foi criado;
key_ - (string) chave do protótipo do item;
lastcheck - (timestamp) hora em que o item foi descoberto pela última vez;
ts_delete - (timestamp) hora em que um item que não é mais descoberto será ser excluído.] [selectPreprocessing|query|Retorna uma propriedade de **pré-processamento** com opções de pré-processamento de item.

Tem as seguintes propriedades:
type - (string) O tipo de opção de pré-processamento:
1 - Multiplicador personalizado;
2 - Aparar à direita;
3 - Aparar à esquerda;
4 - Aparar;
5 - Correspondência de expressão regular;
6 - Booleano para decimal;
7 - Octal para decimal;
8 - Hexadecimal para decimal;
9 - Alteração simples;
10 - Alteração por segundo;
11 - XML XPath;
12 - JSON-Path;
13 - No intervalo;
14 - Corresponde à expressão regular;
15 - Não corresponde à expressão regular;
16 - Verifique se há erro no JSON;
17 - Verificar erro no XML;
18 - Verificar erro usando expressão regular;
19 - Descartar inalterado;
20 - Descartar inalterado com pulsação;
21 - JavaScript;
22 - Padrão Prometheus;
23 - Prometheus para JSON;
24 - CSV para JSON;
25 - Substituir;
26 - Verificar valor não suportado;
27 - XML para JSON.

params - (string) Parâmetros adicionais usados pela opção de pré-processamento. Vários parâmetros são separados pelo caractere LF (\n).
error_handler - (string) Tipo de ação usado em caso de falha na etapa de pré-processamento:
0 - Mensagem de erro é definida pelo servidor Zabbix;
1 - Descartar valor;
2 - Definir valor personalizado;
3 - Definir mensagem de erro personalizada.

error_handler_params - (string) Parâmetros do manipulador de erro.] [selectTags|query|Retorne as tags do item na propriedade **tags**.] [selectValueMap|query|Retorne uma propriedade **valuemap** com o mapa de valor do item.] [filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.

Suporta filtros adicionais:
host - nome técnico do host ao qual o item pertence.] [limitSelects|integer|Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:
selectGraphs - os resultados serão classificados por name;
selectTriggers - resultados será ordenado por descrição.] [sortfield|string/array|Classifique o resultado pelas propriedades fornecidas.

Os valores possíveis são: itemid, name, key_, delay, history, trends, type e status.] [countOutput|boolean|Esses parâmetros sendo comuns

para todos os métodos get são descritos em detalhes na página [comentário de referência](#).| |editável|booleano|^| |exclui-Pesquisa|boolean|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Encontrando itens por chave

Recupere todos os itens do host com ID "10084" que tenham a palavra "sistema" na chave e classifique-os por nome.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.get",
  "parâmetros": {
    "saída": "estender",
    "hostids": "10084",
    "procurar": {
      "sistema chave"
    },
    "sortfield": "nome"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "itemid": "23298",
      "tipo": "0",
      "snmp_oid": "",
      "hostid": "10084",
      "name": "Mudanças de contexto por segundo",
      "key_": "system.cpu.switches",
      "atraso": "1m",
      "história": "7d",
      "tendências": "365d",
      "lastvalue": "2552",
      "último relógio": "1351090998",
      "valor anterior": "2641",
      "estado": "0",
      "estado": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "unidades": "sp",
      "erro": "",
      "logtimefmt": "",
      "templateid": "22680",
      "valuemap": "0",
      "parâmetros": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "nome do usuário": "",
      "senha": "",
      "chave pública": "",
      "chave privada": "",
    }
  ]
}
```

```

    "lastns": "564054253",
    "bandeiras": "0",
    "interfaceid": "1",
    "Descrição": "",
    "link_inventário": "0",
    "vida": "0s",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "tempo limite": "3s",
    "url": "",
    "query_fields": [],
    "Postagens": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "proxy HTTP": "",
    "cabecalhos": [],
    "recuperar_mod0": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "parâmetros": []
},
{
    "itemid": "23299",
    "tipo": "0",
    "snmp_oid": "",
    "hostid": "10084",
    "name": "CPU $ 2 vezes",
    "key_": "system.cpu.util[,idle]",
    "atraso": "1m",
    "história": "7d",
    "tendências": "365d",
    "lastvalue": "86.031879",
    "lastclock": "1351090999",
    "valor anterior": "85.306944",
    "estado": "0",
    "estado": "0",
    "value_type": "0",
    "trapper_hosts": "",
    "unidades": "%",
    "erro": "",
    "logtimefmt": "",
    "modeloid": "17354",
    "valuemap": "0",
    "parâmetros": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "nome do usuário": "",
    "senha": "",
    "chave pública": "",
    "chave privada": "",
    "lastns": "564256864",
    "bandeiras": "0",
    "interfaceid": "1",
    "description": "0 tempo que a CPU passou sem fazer nada.",

```

```

    "link_inventário": "0",
    "vida": "0s",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "tempo limite": "3s",
    "url": "",
    "query_fields": [],
    "Postagens": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "proxy HTTP": "",
    "cabecalhos": [],
    "recuperar_modo": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "parâmetros": []
  },
  {
    "itemid": "23300",
    "tipo": "0",
    "snmp_oid": "",
    "hostid": "10084",
    "name": "CPU $ 2 vezes",
    "key_": "system.cpu.util[,interromper]",
    "história": "7d",
    "tendências": "365d",
    "lastvalue": "0,008389",
    "lastclock": "1351091000",
    "valor anterior": "0,000000",
    "estado": "0",
    "estado": "0",
    "value_type": "0",
    "trapper_hosts": "",
    "unidades": "%",
    "erro": "",
    "logtimefmt": "",
    "modeloid": "22671",
    "valuemap": "0",
    "parâmetros": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "nome do usuário": "",
    "senha": "",
    "chave pública": "",
    "chave privada": "",
    "lastns": "564661387",
    "bandeiras": "0",
    "interfaceid": "1",
    "description": "A quantidade de tempo que a CPU está atendendo a interrupções de hardware.",
    "link_inventário": "0",
    "vida": "0s",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",

```



```

        "tempo limite": "3s",
        "url": "",
        "query_fields": [],
        "Postagens": "",
        "status_codes": "200",
        "follow_redirects": "1",
        "post_type": "0",
        "proxy HTTP": "",
        "cabecalhos": [],
        "recuperar_modulo": "0",
        "request_method": "0",
        "output_format": "0",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "parâmetros": []
    }
],
    "id": 1
}

```

Encontrando itens dependentes por chave

Recupere todos os itens dependentes do host com ID "10116" que tenham o palavra "apache" na chave.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "item.get",
    "parâmetros": {
        "saída": "estender",
        "hostids": "10116",
        "procurar": {
            "key_": "apache"
        },
        "filtro": {
            "tipo": "18"
        }
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": [
        {
            "itemid": "25550",
            "tipo": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "nome": "Dias",
            "key_": "apache.status.uptime.days",
            "atraso": "",
            "história": "90d",
            "tendências": "365d",
            "estado": "0",
            "value_type": "3",
            "trapper_hosts": "",

```

```

    "unidades": "",
    "Fórmula": "",
    "erro": "",
    "logtimefmt": "",
    "templateid": "0",
    "valuemap": "0",
    "parâmetros": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "nome do usuário": "",
    "senha": "",
    "chave pública": "",
    "chave privada": "",
    "bandeiras": "0",
    "interfaceid": "0",
    "Descrição": "",
    "link_inventário": "0",
    "vida": "30d",
    "estado": "0",
    "evaltype": "0",
    "master_itemid": "25545",
    "jmx_endpoint": "",
    "tempo limite": "3s",
    "url": "",
    "query_fields": [],
    "Postagens": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "proxy HTTP": "",
    "cabecalhos": [],
    "recuperar_mod0": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "último relógio": "0",
    "últimos": "0",
    "último valor": "0",
    "valor anterior": "0",
    "parâmetros": []
  },
  {
    "itemid": "25555",
    "tipo": "18",
    "snmp_oid": "",
    "hostid": "10116",
    "nome": "Horas",
    "key_": "apache.status.uptime.hours",
    "atraso": "0",
    "história": "90d",
    "tendências": "365d",
    "estado": "0",
    "value_type": "3",
    "trapper_hosts": "",
    "unidades": "",
    "Fórmula": "",
    "erro": "",

```

```

        "logtimefmt": "",
        "templateid": "0",
        "valuemap": "0",
        "parâmetros": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "nome do usuário": "",
        "senha": "",
        "chave pública": "",
        "chave privada": "",
        "bandeiras": "0",
        "interfaceid": "0",
        "Descrição": "",
        "link_inventário": "0",
        "vida": "30d",
        "estado": "0",
        "evaltype": "0",
        "master_itemid": "25545",
        "jmx_endpoint": "",
        "tempo limite": "3s",
        "url": "",
        "query_fields": [],
        "Postagens": "",
        "status_codes": "200",
        "follow_redirects": "1",
        "post_type": "0",
        "proxy HTTP": "",
        "cabecalhos": [],
        "recuperar_mod0": "0",
        "request_method": "0",
        "output_format": "0",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "último relógio": "0",
        "últimos": "0",
        "último valor": "0",
        "valor anterior": "0",
        "parâmetros": []
    }
],
    "id": 1
}

```

Localizar item do agente HTTP

Encontre o item do agente HTTP com o XML do tipo de corpo da postagem para o ID do host específico.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "item.get",
    "parâmetros": {
        "hostids": "10255",
        "filtro": {
            "tipo": "19",
            "post_type": "3"
        }
    },
    "id": 3,
}

```

```
"auth": "d678e0b85688ce578ff061bd29a20d3b"
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "itemid": "28252",
      "tipo": "19",
      "snmp_oid": "",
      "hostid": "10255",
      "name": "item de modelo",
      "chave_": "ti",
      "atraso": "30s",
      "história": "90d",
      "tendências": "365d",
      "estado": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "unidades": "",
      "Fórmula": "",
      "erro": "",
      "logtimefmt": "",
      "templateid": "0",
      "valuemap": "0",
      "parâmetros": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "nome do usuário": "",
      "senha": "",
      "chave pública": "",
      "chave privada": "",
      "bandeiras": "0",
      "interfaceid": "0",
      "Descrição": "",
      "link_inventário": "0",
      "vida": "30d",
      "estado": "0",
      "evaltype": "0",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "tempo limite": "3s",
      "url": "localhost",
      "campos_consulta": [
        {
          "modo": "xml"
        }
      ],
      "posts": "<body>\r\n<![CDATA[{$MACRO}<foo></bar>]]>\r\n</body>",
      "status_codes": "200",
      "follow_redirects": "0",
      "post_type": "3",
      "proxy HTTP": "",
      "cabecalhos": [],
      "recuperar_mod0": "1",
      "request_method": "3",
      "output_format": "0",
      "ssl_cert_file": "",
      "ssl_key_file": "",
      "ssl_key_password": "",
      "verify_peer": "0",
    }
  ]
}
```

```

        "verify_host": "0",
        "allow_traps": "0",
        "último relógio": "0",
        "últimos": "0",
        "último valor": "0",
        "valor anterior": "0",
        "parâmetros": []
    }
],
    "id": 3
}

```

Recuperando itens com regras de pré-processamento

Recupere todos os itens e suas regras de pré-processamento do host com ID "10254".

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "item.get",
    "parâmetros": {
        "output": ["itemid", "name", "key_"],
        "selectPreprocessing": "estender",
        "hostids": "10254"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "itemid": "23865",
        "name": "Exemplo de agente http JSON",
        "key_": "json",
        "pré-processando": [
            {
                "tipo": "12",
                "params": "$$.random",
                "error_handler": "1",
                "error_handler_params": ""
            }
        ]
    },
    "id": 1
}

```

Veja também

- [Descoberta regra](#)
- [Gráfico](#)
- [Host](#)
- [Hospedeiro interface](#)
- [Trigger](#)

Fonte

CItem::get() em ui/include/classes/api/services/CItem.php.

item.update

Descrição

`object item.update(object/array items)`

Este método permite atualizar itens existentes.

Note:

Itens da web não podem ser atualizados através da API Zabbix.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do item a serem atualizadas.

A propriedade `itemid` deve ser definida para cada item, todos os outros propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas os outros permanecerão inalterados.

Além das **propriedades padrão do item**, o método aceita os seguintes parâmetros.

|Parâmetro|Tipo|Descrição| |-----|-----| |preprocessing|array|Pré-processamento de item
opções para substituir as opções de pré-processamento atuais.| |tags|array|Item tags.|

Valores de retorno

(object) Retorna um objeto contendo os IDs dos itens atualizados sob a propriedade `itemids`.

Exemplos

Habilitando um item

Habilite um item, ou seja, defina seu status como "0".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.atualização",
  "parâmetros": {
    "itemid": "10092",
    "estado": 0
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "10092"
    ]
  },
  "id": 1
}
```

Atualizar item dependente

Atualize o nome do item dependente e o ID do item mestre. Apenas dependências do mesmo host são permitidos, portanto, o item Mestre e Dependente deve ter o mesmo `hostid`.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.atualização",
  "parâmetros": {
    "name": "Nome atualizado do item dependente",
    "master_itemid": "25562",
  },
}
```

```
    "itemid": "189019"
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "189019"
    ]
  },
  "id": 1
}
```

Atualizar item do agente HTTP

Ative a captura de valor do item.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.atualização",
  "parâmetros": {
    "itemid": "23856",
    "allow_traps": "1"
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "23856"
    ]
  },
  "id": 1
}
```

Atualizando um item com pré-processamento

Atualizar um item com a regra de pré-processamento de item "No intervalo".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.atualização",
  "parâmetros": {
    "itemid": "23856",
    "pré-processando": [
      {
        "tipo": "13",
        "params": "\n100",
        "error_handler": "1",
        "error_handler_params": ""
      }
    ]
  },
  "auth": "700ca65537074ec963db7efabda78259",
}
```

```
    "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "23856"
    ]
  },
  "id": 1
}
```

Atualizando um item de script

Atualize um item de script com um script diferente e remova itens desnecessários parâmetros que foram usados pelo script anterior.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "item.atualização",
  "parâmetros": {
    "itemid": "23865",
    "parâmetros": [],
    "script": "Zabbix.Log(3, 'Teste de log');\nreturn 1;"
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "23865"
    ]
  },
  "id": 1
}
```

Fonte

CItem::update() em ui/include/classes/api/services/CItem.php.

Item de gráfico

Esta classe foi projetada para trabalhar com itens de gráfico.

Referências de objeto:

- [item do gráfico](#)

Métodos disponíveis:

- [graphitem.get](#) - recuperando itens do gráfico

> Objeto de item de gráfico

Os objetos a seguir estão diretamente relacionados à API `graphitem`.

Item de gráfico

Note:

Os itens do gráfico só podem ser modificados por meio do graph API.

O objeto de item de gráfico tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |gitemid|string|(readonly) ID do item do gráfico. | **cor**
(obrigatório)|string|A cor de desenho do item do gráfico como um código de cor hexadecimal. | **itemid**
(obrigatório)|string|ID do item. | **calc_fnc**|inteiro|Valor do item que será exibido.

Valores possíveis:
1 - valor mínimo;
2 - (padrão) valor médio;
4 - valor máximo;
7 - todos os valores;
9 - último valor, usado apenas por gráficos de pizza e explodidos. | **drawtype**|inteiro|Desenhe o estilo do item do gráfico.

Valores possíveis:
0 - (padrão) linha;
1 - região preenchida;
2 - linha em negrito;
3 - ponto;
4 - linha tracejada;
5 - linha gradiente. | **graphid**|string|ID do gráfico ao qual o item de gráfico pertence. | **sortorder**|integer|Posição do item no gráfico.

Padrão: começa com 0 e aumenta em um a cada entrada. | **tipo**|inteiro|Tipo de item do gráfico.

Valores possíveis:
0 - (padrão) simples;
2 - soma do gráfico, usado apenas por gráficos de pizza e explodidos. | **yaxisside**|inteiro|Lado do gráfico onde a escala Y do item do gráfico será desenhada.

Valores possíveis:
0 - (padrão) lado esquerdo;
1 - lado direito. |

graphitem.get

Descrição

integer/array graphitem.get(object parameters)

O método permite recuperar itens do gráfico de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----| |graphids|string/array|Retorna apenas itens de gráfico que pertencem aos gráficos fornecidos. | **itemids**|string/array|Retorna apenas itens de gráfico com os IDs de item fornecidos. | **type**|integer|Retorna apenas itens de gráfico com o tipo fornecido.

Consulte a [página de objeto de item de gráfico](#) para obter uma lista de tipos de item de gráfico suportados. | **selectGraphs**|query|Retorne uma propriedade **graphs** com um array de gráficos ao qual o item pertence. | **sortfield**|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: gitemid. | **countOutput**|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página [comentário de referência](#). | **editável**|booleano|^ | **limite**|inteiro|^ | **saída**|consulta|^ | **preservekeys**|boolean|^ | **ordem de classificação**|cadeia/matriz|^ |

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando itens de gráfico de um gráfico

Recupere todos os itens de gráfico usados em um gráfico com informações adicionais sobre o item e o host.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "graphitem.get",
  "parâmetros": {
    "saída": "estender",
    "grafids": "387"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
```

```
    "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "gitemid": "1242",
      "grafid": "387",
      "itemid": "22665",
      "tipo de desenho": "1",
      "ordem de classificação": "1",
      "cor": "FF5555",
      "yaxisside": "0",
      "calc_fnc": "2",
      "tipo": "0",
      "key_": "system.cpu.util[,roubar]",
      "hostid": "10001",
      "bandeiras": "0",
      "host": "Linux"
    },
    {
      "gitemid": "1243",
      "grafid": "387",
      "itemid": "22668",
      "tipo de desenho": "1",
      "ordem de classificação": "2",
      "cor": "55FF55",
      "yaxisside": "0",
      "calc_fnc": "2",
      "tipo": "0",
      "key_": "system.cpu.util[,softirq]",
      "hostid": "10001",
      "bandeiras": "0",
      "host": "Linux"
    },
    {
      "gitemid": "1244",
      "grafid": "387",
      "itemid": "22671",
      "tipo de desenho": "1",
      "ordem de classificação": "3",
      "cor": "009999",
      "yaxisside": "0",
      "calc_fnc": "2",
      "tipo": "0",
      "key_": "system.cpu.util[,interromper]",
      "hostid": "10001",
      "bandeiras": "0",
      "host": "Linux"
    }
  ],
  "id": 1
}
```

Veja também

- [Gráfico](#)

Fonte

CGraphItem::get() em ui/include/classes/api/services/CGraphItem.php.

Macro do usuário

Esta classe foi projetada para trabalhar com macros de host e globais.

Referências de objetos:

- [Macro global](#)
- [Host macro](#)

Métodos disponíveis:

- [usermacro.create](#) - criando novas macros de host
- [usermacro.createglobal](#) - criando novas macros globais
- [usermacro.delete](#) - excluindo macros de host
- [usermacro.deleteglobal](#) - excluindo macros globais
- [usermacro.get](#) - recuperando host e macros globais
- [usermacro.update](#) - atualizando macros de host
- [usermacro.updateglobal](#) - atualizando macros globais

> Objeto macro de usuário

Os objetos a seguir estão diretamente relacionados ao `usermacro` API.

Global macro

O objeto macro global tem as seguintes propriedades.

Propriedade	Tipo	Descrição
globalmacroid	string	(readonly) ID do macro global.
macro (requisitado)	string	Macro string.
valor (requisitado)	string	Valor do macro.
tipo	integer	Tipo do macro. Valores possíveis: 0 - (padrão) Texto macro; 1 - Macro seguro; 2 - Vault seguro.
descrição	string	Descrição do macro.

Note que para alguns métodos (update, delete) a combinação de parâmetro requisitada/opcional é diferente.

Host macro

O objeto do host macro define a disponibilidade de um macro no host, prototype host ou template. Ele tem as seguintes propriedades:

Propriedade	Tipo	Descrição
hostmacroid	string	(readonly) ID do host macro.
hostid (requisitado)	string	ID do macro que pertence ao host.
macro (requisitado)	string	Macro string.
value (requisitado)	string	Valor do macro.
tipo	integer	Tipo do macro. Valores Possíveis: 0 - (default) Texto macro; 1 - Macro seguro; 2 - Vault seguro.

Propriedade	Tipo	Descrição
Descrição	string	Descrição do macro.

Note que para alguns dos métodos (update, delete) a combinação do parâmetro requisitado/opcional é diferente.

usermacro.create

Descrição

object usermacro.create(object/array hostMacros)

Este método permite criar novas macros de host.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(objeto/array) Macros de host a serem criadas.

O método aceita macros de host com as **propriedades de macro de host padrão**.

Valores de retorno

(object) Retorna um objeto contendo os IDs das macros de host criadas sob a propriedade hostmacroids. A ordem dos IDs retornados corresponde à ordem das macros de host passadas.

Exemplos

Criando uma macro de host

Crie uma macro de host "{\$SNMP_COMMUNITY}" com o valor "public" no host "10198".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.create",
  "params": {
    "hostid": "10198",
    "macro": "{$SNMP_COMMUNITY}",
    "value": "public"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "hostmacroids": [
      "11"
    ]
  },
  "id": 1
}
```

Fonte

CUserMacro::create() in ui/include/classes/api/services/CUserMacro.php.

usermacro.createglobal

Descrição

`object usermacro.createglobal(object/array globalMacros)`

Este método permite criar novas macros globais.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Macros globais a serem criadas.

O método aceita macros globais com as [propriedades de macros globais padrão](#).

Valores de retorno

(object) Retorna um objeto contendo os IDs das macros globais criadas sob a propriedade `globalmacroids`. A ordem dos IDs retornados corresponde à ordem das macros globais passadas.

Exemplos

Criando uma macro global

Crie uma macro global `"{$SNMP_COMMUNITY}"` com o valor `"public"`.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.createglobal",
  "params": {
    "macro": "{$SNMP_COMMUNITY}",
    "value": "public"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "globalmacroids": [
      "6"
    ]
  },
  "id": 1
}
```

Fonte

`CUserMacro::createGlobal()` in `ui/include/classes/api/services/CUserMacro.php`.

usermacro.delete

Description

`object usermacro.delete(array hostMacroIds)`

This method allows to delete host macros.

Note:

This method is only available to Admin and Super admin user types. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parâmetros

(array) IDs das macros de hosts a serem deletadas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das macros de host excluídas na propriedade `hostmacroids`.

Exemplos

Excluindo várias macros de host

Exclua duas macros de host.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.delete",
  "params": [
    "32",
    "11"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "hostmacroids": [
      "32",
      "11"
    ]
  },
  "id": 1
}
```

Fonte

`CUserMacro::delete()` in `ui/include/classes/api/services/CUserMacro.php`.

usermacro.deleteglobal

Description

`object usermacro.deleteglobal(array globalMacroIds)`

This method allows to delete global macros.

Note:

This method is only available to Super admin user type. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parameters

(array) IDs of the global macros to delete.

Return values

(object) Returns an object containing the IDs of the deleted global macros under the `globalmacroids` property.

Exemplos

Deleting multiple global macros

Delete two global macros.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.deleteglobal",
  "params": [
```

```

        "32",
        "11"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "globalmacroids": [
            "32",
            "11"
        ]
    },
    "id": 1
}

```

Fonte

CUserMacro::deleteGlobal() in ui/include/classes/api/services/CUserMacro.php.

usermacro.get

Descrição

integer/array usermacro.get(object parameters)

O método permite recuperar macros de host e globais de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [funções do usuário](#) para obter mais informações.

Parameters

(object) Parameters defining the desired output.

The method supports the following parameters.

Parameter	Type	Description
globalmacro	flag	Return global macros instead of host macros.
globalmacroids	string/array	Return only global macros with the given IDs.
groupids	string/array	Return only host macros that belong to hosts or templates from the given host groups.
hostids	string/array	Return only macros that belong to the given hosts or templates.
hostmacroids	string/array	Return only host macros with the given IDs.

Parameter	Type	Description
inherited	boolean	If set to true return only host prototype user macros inherited from a template.
selectGroups	query	Return host groups that the host macro belongs to in the groups property.
selectHosts	query	Used only when retrieving host macros. Return hosts that the host macro belongs to in the hosts property.
selectTemplates	query	Used only when retrieving host macros. Return templates that the host macro belongs to in the templates property.
sortfield	string/array	Used only when retrieving host macros. Sort the result by the given properties.
countOutput	boolean	Possible value: macro. These parameters being common for all <code>get</code> methods are described in detail in the reference commentary page.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Return values

(integer/array) Returns either:

- an array of objects;

- the count of retrieved objects, if the countOutput parameter has been used.

Exemplos

Retrieving host macros for a host

Retrieve all host macros defined for host "10198".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.get",
  "params": {
    "output": "extend",
    "hostids": "10198"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostmacroid": "9",
      "hostid": "10198",
      "macro": "{$INTERFACE}",
      "value": "eth0",
      "description": "",
      "type": "0"
    },
    {
      "hostmacroid": "11",
      "hostid": "10198",
      "macro": "{$SNMP_COMMUNITY}",
      "value": "public",
      "description": "",
      "type": "0"
    }
  ],
  "id": 1
}
```

Retrieving global macros

Retrieve all global macros.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.get",
  "params": {
    "output": "extend",
    "globalmacro": true
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
```

```

        "globalmacroid": "6",
        "macro": "{$SNMP_COMMUNITY}",
        "value": "public",
        "description": "",
        "type": "0"
    }
],
    "id": 1
}

```

Fonte

CUserMacro::get() in ui/include/classes/api/services/CUserMacro.php.

usermacro.update

Description

object usermacro.update(object/array hostMacros)

This method allows to update existing host macros.

Note:

This method is only available to Admin and Super admin user types. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parameters

(object/array) **Host macro properties** to be updated.

The `hostmacroid` property must be defined for each host macro, all other properties are optional. Only the passed properties will be updated, all others will remain unchanged.

Return values

(object) Returns an object containing the IDs of the updated host macros under the `hostmacroids` property.

Exemplos

Changing the value of a host macro

Change the value of a host macro to "public".

Request:

```

{
    "jsonrpc": "2.0",
    "method": "usermacro.update",
    "params": {
        "hostmacroid": "1",
        "value": "public"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "hostmacroids": [
            "1"
        ]
    },
    "id": 1
}

```

Fonte

CUserMacro::update() in ui/include/classes/api/services/CUserMacro.php.

usermacro.updateglobal

Descrição

object usermacro.updateglobal(object/array globalMacros)

Este método permite atualizar macros globais existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Para mais informações, consulte [User roles](#).

Parâmetros

Propriedades (object/array) **Global macro properties** a serem atualizadas. A propriedade globalmacroid deve ser definida para cada macro global, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Valores retornados

(object) Retorna um objeto contendo a ID dos macros globais atualizados dentro da propriedade globalmacroids .

Exemplos

Alterando o valor de um macro global

Altere o valor de um macro global para "público".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "usermacro.updateglobal",
  "params": {
    "globalmacroid": "1",
    "value": "public"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "globalmacroids": [
      "1"
    ]
  },
  "id": 1
}
```

Fonte

CUserMacro::updateGlobal() in ui/include/classes/api/services/CUserMacro.php.

Manutenção

Esta classe é projetada para trabalhar com manutenções.

Referências de objetos:

- [Maintenance](#)
- [Time period](#)

Métodos disponíveis:

- **maintenance.create** - criando novas manutenções
- **maintenance.delete** - excluindo manutenção
- **maintenance.get** - recuperando manutenção
- **maintenance.update** - atualização de manutenção

> Objeto de manutenção

Os objetos a seguir estão diretamente relacionados à API **maintenance**.

Manutenção

O objeto de manutenção tem as seguintes propriedades.

Propriedade	Tipo	Descrição
maintenanceid	string	(somente leitura) ID da manutenção.
name (required)	string	Nome da manutenção.
active_since (requerido)	timestamp	Hora em que a manutenção se torna ativa.
active_till (requerido)	timestamp	O valor fornecido será arredondado para minutos. Hora em que a manutenção deixa de estar ativa.
description	string	O valor fornecido será arredondado para minutos. Descrição da manutenção.
maintenance_type	integer	Tipo de manutenção.
tags_evaltype	integer	Valores possíveis: 0 - (padrão) com coleta de dados; 1 - sem coleta de dados. Método de avaliação da tag do problema.
		Valores possíveis: 0 - (padrão) And/Or; 2 - Or.

Período de tempo

O objeto período de tempo é utilizado para definir os períodos em que a manutenção deve entrar em vigor. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
period	integer	Duração do período de manutenção em segundos.
		O valor fornecido será arredondado para minutos.
timeperiod_type	integer	Padrão: 3600. Tipo de período de tempo.
		Valores possíveis: 0 - (padrão) apenas uma vez; 2 - diariamente; 3 - semanalmente; 4 - mensalmente.

Propriedade	Tipo	Descrição
start_date	timestamp	<p>Data em que o período de manutenção deve entrar em vigor.</p> <p>Usado apenas para períodos de tempo.</p> <p>O valor fornecido será arredondado para minutos.</p>
start_time	integer	<p>Padrão: data atual.</p> <p>Hora do dia em que a manutenção começa em segundos.</p> <p>Usado para períodos diários, semanais e mensais.</p> <p>O valor fornecido será arredondado para minutos.</p>
every	integer	<p>Padrão: 0.</p> <p>Usado para períodos diários, semanais e mensais.</p> <p>Para períodos diários e semanais, <code>every</code> define intervalos de dias ou semanas em que a manutenção deve entrar em vigor.</p> <p>Padrão: 1.</p> <p>Para períodos mensais, se a propriedade <code>dayofweek</code> contiver pelo menos um dia da semana selecionado, a propriedade <code>every</code> definirá a semana do mês em que a manutenção deve entrar em vigor.</p> <p>Valores possíveis:</p> <ul style="list-style-type: none"> 1 - (padrão) primeira semana; 2 - segunda semana; 3 - terceira semana; 4 - quarta semana; 5 - última semana.
dayofweek	integer	<p>Dias da semana em que a manutenção deve entrar em vigor.</p> <p>Os dias são armazenados em formato binário com cada bit representando o dia correspondente. Por exemplo, 4 é igual a 100 em binário e significa que a manutenção será habilitada na quarta-feira.</p> <p>Usado para períodos semanais e mensais. Necessário apenas para períodos semanais.</p>
day	integer	<p>Pelo menos um <code>dayofweek</code> ou <code>day</code> deve ser especificado para períodos mensais.</p> <p>Dia do mês em que a manutenção deve entrar em vigor.</p> <p>Usado apenas para períodos mensais.</p>
month	integer	<p>Pelo menos um <code>dayofweek</code> ou <code>day</code> deve ser especificado para períodos mensais.</p> <p>Meses em que a manutenção deve entrar em vigor.</p> <p>Os meses são armazenados em formato binário com cada bit representando o mês correspondente. Por exemplo, 5 é igual a 101 em binário e significa que a manutenção será ativada em janeiro e março.</p> <p>Necessário apenas para períodos mensais.</p>

Tag do problema

O objeto tag de problema é utilizado para definir quais problemas devem ser suprimidos quando a manutenção entrar em vigor. Ele tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tag (required)	string	Nome da tag do problema.
operator	integer	Operador de condição. Valores possíveis: 0 - Igual a; 2 - (padrão) Contém.
value	string	Valor da tag do problema.

Tags só podem ser especificadas para períodos de manutenção com coleta de dados ("maintenance_type":0).

maintenance.create

Descrição

`object maintenance.create(object/array maintenances)`

Este método permite criar novas manutenções.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Manutenções a criar.

Além das **propriedades de manutenção padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
groups	object/array	Host grupos que passará por manutenção. Os grupos de hosts devem ter a propriedade <code>groupid</code> definida.
hosts	object/array	Pelo menos um objeto de <code>groups</code> ou <code>hosts</code> deve ser especificado. Hosts que passará por manutenção. Os hosts devem ter a propriedade <code>hostid</code> definida.
timeperiods (requerido)	object/array	Pelo menos um objeto de <code>groups</code> ou <code>hosts</code> deve ser especificado. Manutenção time periods .
tags	object/array	Problem tags . Defina quais problemas devem ser suprimidos. Se nenhuma tag for fornecida, todos os problemas de host de manutenção ativos serão suprimidos.

Valores de retorno

(object) Retorna um objeto contendo os IDs das manutenções criadas na propriedade `maintenanceids`. A ordem dos IDs retornados corresponde à ordem da manutenção aprovada.

Exemplos

Criando uma manutenção

Crie uma manutenção com coleta de dados para o grupo de hosts com ID "2" e com tags de problema **service:mysqlid** e **error**. Deve estar ativo de 22.01.2013 a 22.01.2014, entrar em vigor todos os domingos às 18h e durar uma hora

Request:

```
{
  "jsonrpc": "2.0",
  "method": "maintenance.create",
  "params": {
    "name": "Sunday maintenance",
    "active_since": 1358844540,
    "active_till": 1390466940,
    "tags_evaltype": 0,
    "groups": [
      {"groupid": "2"}
    ],
    "timeperiods": [
      {
        "period": 3600,
        "timeperiod_type": 3,
        "start_time": 64800,
        "every": 1,
        "dayofweek": 64
      }
    ],
    "tags": [
      {
        "tag": "service",
        "operator": "0",
        "value": "mysqld"
      },
      {
        "tag": "error",
        "operator": "2",
        "value": ""
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "maintenanceids": [
      "3"
    ]
  },
  "id": 1
}
```

Veja também

- [Time period](#)

Fonte

CMaintenance::create() in ui/include/classes/api/services/CMaintenance.php.

maintenance.delete

Descrição

object maintenance.delete(array maintenanceIds)

Este método permite excluir períodos de manutenção.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Para mais informações, consulte [User roles](#).

Parâmetros

(array) IDs dos períodos de manutenção a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos períodos de manutenção excluídos sob a propriedade `maintenanceids`.

Exemplos

Excluindo vários períodos de manutenção

Exclua dois períodos de manutenção.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "maintenance.delete",
  "params": [
    "3",
    "1"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "maintenanceids": [
      "3",
      "1"
    ]
  },
  "id": 1
}
```

Fonte

`CMaintenance::delete()` in `ui/include/classes/api/services/CMaintenance.php`.

maintenance.get**Descrição**

`integer/array maintenance.get(object parameters)`

O método permite recuperar manutenções de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
groupids	string/array	Retornar apenas as manutenções atribuídas aos grupos de hosts fornecidos.
hostids	string/array	Retornar apenas as manutenções atribuídas aos hosts fornecidos.
maintenanceids	string/array	Devolva apenas manutenções com os IDs fornecidos.
selectGroups	query	Retorne uma propriedade groups com grupos de hosts atribuídos à manutenção.
selectHosts	query	Retorne uma propriedade hosts com hosts atribuídos à manutenção.
selectTags	query	Retorna uma propriedade tags com tags de problema da manutenção.
selectTimeperiods	query	Retorna uma propriedade tags com tags de problema da manutenção.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
Os valores possíveis são: maintenanceid, name e maintenance_type.		
countOutput	boolean	Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes em comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna ou:

- uma série de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando manutenção

Recupere todas as manutenções configuradas e os dados sobre os grupos de hosts atribuídos, períodos de tempo definidos e tags de problemas.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "maintenance.get",
  "params": {
    "output": "extend",
    "selectGroups": "extend",
    "selectTimeperiods": "extend",
    "selectTags": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "maintenanceid": "3",
      "name": "Sunday maintenance",
      "maintenance_type": "0",

```

```

        "description": "",
        "active_since": "1358844540",
        "active_till": "1390466940",
        "tags_evaltype": "0",
        "groups": [
            {
                "groupid": "4",
                "name": "Zabbix servers",
                "internal": "0"
            }
        ],
        "timeperiods": [
            {
                "timeperiod_type": "3",
                "every": "1",
                "month": "0",
                "dayofweek": "1",
                "day": "0",
                "start_time": "64800",
                "period": "3600",
                "start_date": "2147483647"
            }
        ],
        "tags": [
            {
                "tag": "service",
                "operator": "0",
                "value": "mysqld",
            },
            {
                "tag": "error",
                "operator": "2",
                "value": ""
            }
        ]
    },
    "id": 1
}

```

Veja também

- [Host](#)
- [Host group](#)
- [Time period](#)

Fonte

CMaintenance::get() in ui/include/classes/api/services/CMaintenance.php.

maintenance.update

Descrição

object maintenance.update(object/array maintenances)

Este método permite atualizar as manutenções existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades de manutenção a serem atualizadas.

A propriedade `maintenanceid` deve ser definida para cada manutenção, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das **propriedades de manutenção padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
groups	object/array	Host grupos para substituir os grupos atuais.
hosts	object/array	Os grupos de hosts devem ter a propriedade <code>groupid</code> definida. Hosts para substituir os hosts atuais.
timeperiods	object/array	Os hosts devem ter a propriedade <code>hostid</code> definida Manutenção períodos de tempo para substituir os períodos atuais.
tags	object/array	Tags de problema para substituir as tags atuais.

Attention:

Pelo menos um host ou grupo de hosts deve ser definido para cada manutenção.

Valores de retorno

(object) Retorna um objeto contendo os IDs das manutenções atualizadas na propriedade `maintenanceids`.

Exemplos

Atribuindo hosts diferentes

Substitua os hosts atualmente atribuídos à manutenção por dois diferentes.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "maintenance.update",
  "params": {
    "maintenanceid": "3",
    "hosts": [
      {"hostid": "10085"},
      {"hostid": "10084"}
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "maintenanceids": [
      "3"
    ]
  },
  "id": 1
}
```

Veja também

- **Time period**

Fonte

`CMaintenance::update()` in `ui/include/classes/api/services/CMaintenance.php`.

Mapa

Esta classe foi projetada para trabalhar com mapas.

Referências de objetos:

- [Map](#)
- [Map element](#)
- [Map link](#)
- [Map URL](#)
- [Map user](#)
- [Map user group](#)
- [Map shape](#)
- [Map line](#)

Métodos disponíveis:

- [map.create](#) - criar novos mapas
- [map.delete](#) - excluir mapas
- [map.get](#) - recuperar mapas
- [map.update](#) - atualizar mapas

> Objeto de mapa

Os objetos a seguir estão diretamente relacionados à API `map`.

Mapa

O objeto de mapa tem as seguintes propriedades.

Propriedade	Tipo	Descrição
<code>sysmapid</code>	string	(somente leitura) ID of the map.
<code>height</code> (requerido)	integer	Altura do mapa em pixels.
<code>name</code> (requerido)	string	Nome do mapa.
<code>width</code> (requerido)	integer	Largura do mapa em pixels.
<code>backgroundid</code>	string	ID da imagem usada como plano de fundo para o mapa.
<code>expand_macros</code>	integer	Se as macros devem ser expandidas em rótulos ao configurar o mapa. Valores possíveis: 0 - (padrão) não expande macros; 1 - expande macros.
<code>expandproblem</code>	integer	Se o acionador do problema será exibido para elementos com um único problema. Valores possíveis: 0 - sempre exibe o número de problemas; 1 - (padrão) exibe o acionador do problema se Há apenas um problema.
<code>grid_align</code>	integer	Se deve ativar o alinhamento da grade. Valores possíveis: 0 - desabilita o alinhamento da grade; 1 - (padrão) habilita o alinhamento da grade.
<code>grid_show</code>	integer	Se deve mostrar a grade no mapa. Valores possíveis: 0 - não mostrar a grade; 1 - (padrão) mostrar a grade.
<code>grid_size</code>	integer	Tamanho da grade do mapa em pixels. Valores compatíveis: 20, 40, 50, 75 e 100. Padrão: 50.

Propriedade	Tipo	Descrição
highlight	integer	Se o realce do ícone está ativado. Valores possíveis: 0 - realce desativado; 1 - (padrão) realce ativado.
iconmapid	string	ID do mapa de ícones usado no mapa.
label_format	integer	Se deve ativar marcadores avançados. Valores possíveis: 0 - (padrão) desativar marcadores avançados; 1 - ativar marcadores avançados.
label_location	integer	Localização do rótulo do elemento do mapa. Valores possíveis: 0 - (padrão) inferior; 1 - esquerda; 2 - direita; 3 - superior.
label_string_host	string	Rótulo personalizado para elementos do host.
label_string_hostgroup	string	Obrigatório para mapas com tipo de rótulo do host personalizado. Rótulo personalizado para elementos do grupo de hosts.
label_string_image	string	Obrigatório para mapas com tipo de rótulo de grupo de hosts personalizado. Rótulo personalizado para elementos de imagem.
label_string_map	string	Obrigatório para mapas com tipo de rótulo de imagem personalizado. Rótulo personalizado para elementos de mapa.
label_string_trigger	string	Obrigatório para mapas com tipo de rótulo de mapa personalizado. Rótulo personalizado para elementos de gatilho.
label_type	integer	Obrigatório para mapas com tipo de rótulo de gatilho personalizado. Tipo de rótulo do elemento do mapa. Valores possíveis: 0 - rótulo; 1 - endereço IP; 2 - (padrão) nome do elemento; 3 - somente status; 4 - nada.
label_type_host	integer	Tipo de rótulo para elementos de host. Valores possíveis: 0 - rótulo; 1 - endereço IP; 2 - (padrão) nome do elemento; 3 - somente status ; 4 - nada; 5 - personalizado.
label_type_image	integer	Tipo de rótulo para elementos do grupo de hosts. Valores possíveis: 0 - rótulo; 2 - (padrão) nome do elemento; 4 - nada; 5 - personalizado.

Propriedade	Tipo	Descrição
label_type_map	integer	Tipo de rótulo para elementos do mapa. Valores possíveis: 0 - rótulo; 2 - (padrão) nome do elemento; 3 - somente status; 4 - nada; 5 - personalizado.
label_type_trigger	integer	Tipo de rótulo para elementos de gatilho. Valores possíveis: 0 - rótulo; 2 - (padrão) nome do elemento; 3 - somente status; 4 - nada; 5 - personalizado.
markelements	integer	Se deve destacar elementos do mapa que mudaram recentemente de status. Valores possíveis: 0 - (padrão) não destacar elementos; 1 - destacar elementos.
severity_min	integer	Gravidade mínima dos acionadores que serão exibidos no mapa. Consulte a propriedade "severity" do acionador para obter uma lista de gravidades de acionadores compatíveis.
show_unack	integer	Como os problemas devem ser exibidos. Valores possíveis: 0 - (padrão) exibe a contagem de todos os problemas; 1 - exibe apenas a contagem de problemas não confirmados; 2 - exibir a contagem de problemas reconhecidos e não reconhecidos separadamente.
userid	string	ID de usuário do proprietário do mapa.
private	integer	Tipo de compartilhamento de mapa. Valores possíveis: 0 - mapa público; 1 - (padrão) mapa privado.
show_suppressed	integer	Se os problemas suprimidos são mostrados. Valores possíveis: 0 - (padrão) ocultar problemas suprimidos; 1 - mostrar problemas suprimidos.

Elemento do mapa

O objeto de elemento de mapa define um objeto exibido em um mapa. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
selementid	string	(somente leitura) ID do elemento do mapa.
elementos (obrigatório)	array	Objeto de dados do elemento. Necessário para elementos de host, grupo de hosts, acionador e tipo de mapa.

Propriedade	Tipo	Descrição
tipo de elemento (obrigatório)	inteiro	Tipo de elemento do mapa. Valores possíveis: 0 - host; 1 - mapa; 2 - trigger; 3 - grupo de host; 4 - imagem.
iconid_off (obrigatório)	string	ID da imagem usada para exibir o elemento no estado padrão.
tipo de área	inteiro	Como os hosts do grupo de hosts separados devem ser exibidos. Valores possíveis: 0 - (padrão) o elemento de grupo de host ocupará todo o mapa; 1 - o elemento do grupo de hosts terá um tamanho fixo.
subtipo de elemento	inteiro	Como um elemento de grupo de hosts deve ser exibido em um mapa. Valores possíveis: 0 - (padrão) exibe o grupo de hosts como um único elemento; 1 - exibe cada host no grupo separadamente.
tipo de avaliação	inteiro	Método de avaliação da condição de filtragem da tag do elemento no mapa. Valores possíveis: 0 - (padrão) E / OU; 2 - OU.
altura	inteiro	Altura do elemento do grupo de hosts de tamanho fixo em pixels. Padrão: 200.
iconid_disabled	string	ID da imagem usada para exibir elementos de mapa desativados. Não utilizado para elementos de imagem.
iconid_maintenance	string	ID da imagem usada para exibir os elementos do mapa em manutenção. Não utilizado para elementos de imagem.
iconid_on	string	ID da imagem usada para exibir os elementos do mapa com problemas. Não utilizado para elementos de imagem.
rótulo	string	Rótulo do elemento.
rótulo_local	inteiro	Localização do rótulo do elemento do mapa. Valores possíveis: -1 - (padrão) localização padrão; 0 - fundo; 1 - esquerda; 2 - direita; 3 - topo.
permissão	inteiro	Tipo de nível de permissão. Valores possíveis: -1 - nenhum; 2 - somente leitura; 3 - leitura-escrita.
sysmapid	string	(somente leitura) ID do mapa ao qual o elemento pertence.
URLs	array	URLs do elemento do mapa.
use_iconmap	inteiro	O objeto URL do elemento do mapa está descrito em detalhes abaixo . Se o mapeamento de ícones deve ser usado para elementos hospedeiros. Valores possíveis: 0 - não use mapeamento de ícones; 1 - (padrão) usa o mapeamento de ícones.

Propriedade	Tipo	Descrição
tipo de visão	inteiro	Algoritmo de posicionamento do elemento do grupo de hosts.
largura	inteiro	Valores possíveis: 0 - (padrão) grade. Largura do elemento do grupo de hosts de tamanho fixo em pixels.
x	inteiro	Padrão: 200. Coordenadas X do elemento em pixels.
y	inteiro	Padrão: 0. Coordenadas Y do elemento em pixels.
		Padrão: 0.

Elemento do mapa - Host

O objeto Host do elemento do mapa define um elemento host.

Propriedade	Tipo	Descrição
hostid	string	ID do Host

Elemento do mapa - Grupo de Hosts

O objeto do grupo de hosts do elemento do mapa define um elemento do grupo de hosts.

Propriedade	Tipo	Descrição
ID de grupo	string	ID do grupo de hosts

Elemento do mapa - Mapa

Define um elemento de mapa.

Propriedade	Tipo	Descrição
sysmapid	string	ID do Mapa

Elemento do Mapa - Trigger

Define um ou mais elementos acionadores.

Propriedade	Tipo	Descrição
Trigger ID	string	Trigger ID

Elemento do Mapa - Etiqueta

O objeto de etiquetas possui as seguintes propriedades.

Propriedade	Tipo	Descrição
etiqueta (obrigatório)	string	Nome da etiqueta do elemento do mapa.

Propriedade	Tipo	Descrição
operador	string	Operador de condição de tag de elemento de mapa. Valores possíveis: 0 - (padrão) Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
valor	string	Valor da etiqueta do elemento do mapa.

Elemento do Mapa - URL

O objeto de URL do elemento de mapa define um link clicável que estará disponível para um elemento de mapa específico. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmapelementurlid	string	(somente leitura) ID da URL do elemento do mapa.
nome (obrigatório)	string	Legenda do link.
url (obrigatório)	string	URL do Link.
selementid	string	ID do elemento do mapa ao qual o URL pertence.

Link do mapa

O objeto de link de mapa define um link entre dois elementos de mapa. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
Link ID	string	(somente leitura) ID do link do mapa.
selementid1 (obrigatório)	string	ID do primeiro elemento do mapa vinculado em uma extremidade.
selementid2 (obrigatório)	string	ID do primeiro elemento do mapa vinculado na outra extremidade.
cor	string	Cor da linha como um código de cor hexadecimal. Padrão: 000000.
tipo de desenho	inteiro	Estilo de desenho de linha de link. Valores possíveis: 0 - (padrão) linha; 2 - linha em negrito; 3 - linha pontilhada; 4 - linha tracejada.
rótulo	string	Rótulo de link.
acionadores de triggers	array	Mapeia acionadores de link para usar como indicadores de status de link.
permissão	inteiro	O objeto acionador do link do mapa é descrito em detalhes abaixo . Tipo de nível de permissão. Valores possíveis: -1 - nenhum; 2 - somente leitura; 3 - leitura-escrita.
sysmapid	string	ID do mapa ao qual o link pertence.

Trigger de link do mapa

O objeto trigger de link de mapa define um indicador de status de link de mapa com base no estado de uma trigger. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
linktriggerid	string	(somente leitura) ID da trigger do link do mapa.
triggerid	string	ID da trigger usada como um indicador de link.
(obrigatório)		
cor	string	Cor do indicador como um código de cor hexadecimal.
		Padrão: DD0000.
tipo de desenho	inteiro	Estilo de desenho do indicador.
		Valores possíveis: 0 - (padrão) linha; 2 - linha em negrito; 3 - linha pontilhada; 4 - linha tracejada.
linkid	string	ID do link do mapa ao qual a trigger do link pertence.

URL do mapa

O objeto de URL do mapa define um link clicável que estará disponível para todos os elementos de um tipo específico no mapa. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmapurlid	string	(somente leitura) ID da URL do mapa.
nome	string	Legenda do link.
(obrigatório)		
URL	string	URL do Link.
(obrigatório)		
tipo de elemento	inteiro	Tipo de elemento do mapa para o qual a URL estará disponível.
		Consulte a propriedade "tipo" do elemento do mapa para obter uma lista de tipos suportados.
		Padrão: 0.
sysmapid	string	ID do mapa ao qual a URL pertence.

Mapa de usuário

Lista de permissões de mapa com base em usuários. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmapuserid	string	(somente leitura) ID do usuário do mapa.
ID do usuário	string	ID do usuário.
(obrigatório)		
permissão	inteiro	Tipo de nível de permissão.
(obrigatório)		
		Valores possíveis: 2 - somente leitura; 3 - leitura-escrita;

Mapa de grupo de usuário

Lista de permissões de mapa com base em grupos de usuários. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmapusrgrpid	string	(somente leitura) ID do grupo de usuários do mapa.

Propriedade	Tipo	Descrição
usrgrpId (obrigatório)	string	ID do grupo de usuário.
permissão (obrigatório)	inteiro	Tipo de nível de permissão. Valores possíveis: 2 - somente leitura; 3 - leitura-escrita;

Formas de mapa

O objeto de forma de mapa define uma forma geométrica (com ou sem texto) exibida em um mapa. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmap_shapeId	string	(somente leitura) ID do elemento de forma do mapa.
tipo (obrigatório)	inteiro	Tipo de elemento de forma de mapa. Valores possíveis: 0 - retângulo; 1 - elipse.
x	inteiro	A propriedade é necessária quando novas formas são criadas. X-Coordenadas X da forma em pixels.
y	inteiro	Padrão: 0. Y-Coordenadas Y da forma em pixels.
largura	inteiro	Padrão: 0. Largura da forma em pixels.
altura	inteiro	Padrão: 200. Altura da forma em pixels.
texto	string	Padrão: 200. Text da forma.
fonte	inteiro	Fonte do texto dentro da forma. Valores possíveis: 0 - Georgia, serif 1 - "Palatino Linotype", "Book Antiqua", Palatino, serif 2 - "Times New Roman", Times, serif 3 - Arial, Helvetica, sans-serif 4 - "Arial Black", Gadget, sans-serif 5 - "Comic Sans MS", cursive, sans-serif 6 - Impact, Charcoal, sans-serif 7 - "Lucida Sans Unicode", "Lucida Grande", sans-serif 8 - Tahoma, Geneva, sans-serif 9 - "Trebuchet MS", Helvetica, sans-serif 10 - Verdana, Geneva, sans-serif 11 - "Courier New", Courier, monospace 12 - "Lucida Console", Monaco, monospace
tamanho da fonte	inteiro	Padrão: 9. Tamanho da fonte em pixels.
cor da fonte	string	Padrão: 11. Cor da fonte. Padrão: '000000'.

Propriedade	Tipo	Descrição
alinhamento horizontal do texto	inteiro	Alinhamento horizontal do texto. Valores possíveis: 0 - centro; 1 - esquerda; 2 - direita.
alinhamento vertical do texto	inteiro	Padrão: 0. Alinhamento vertical do texto. Valores possíveis: 0 - meio; 1 - topo; 2 - inferior.
tipo de borda	inteiro	Padrão: 0. Tipo de borda. Valores possíveis: 0 - nenhum; 1 - _____; 2 - - - -; 3 - - - - -.
largura da borda	inteiro	Padrão: 0. Largura da borda em pixels.
cor da borda	string	Padrão: 0. Cor da borda.
cor de fundo	string	Padrão: '000000'. Cor de fundo (cor de preenchimento).
zindex	inteiro	Padrão: (vazio). Valor usado para ordenar todas as formas e linhas (z-index). Padrão: 0.

Linhas do mapa

O objeto de linha do mapa define uma linha exibida em um mapa. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
sysmap_shapeid	string	(somente leitura) ID do elemento de forma do mapa.
x1	inteiro	Coordenadas X do ponto de linha 1 em pixels. Padrão: 0.
y1	inteiro	Coordenadas Y do ponto de linha 1 em pixels. Padrão: 0.
x2	inteiro	Coordenadas X do ponto de linha 2 em pixels. Padrão: 200.
y2	inteiro	Coordenadas Y do ponto de linha 2 em pixels. Padrão: 200.

Propriedade	Tipo	Descrição
tipo de linha	inteiro	Tipo das linhas. Valores possíveis: 0 - nenhum; 1 - _____; 2 - - -; 3 - - - - -.
espessura da linha	inteiro	Padrão: 0. Largura das linhas em pixels.
cor da linha	string	Padrão: 0. Cor da linha.
zindex	inteiro	Padrão: '000000'. Valor usado para ordenar todas as formas e linhas (z-index). Padrão: 0.

map.create

Descrição

`object map.create(object/array maps)`

Este método permite criar novos mapas.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Mapas para criar.

Além das [propriedades padrão do mapa](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
links	array	Mapa links a ser criado no mapa.
selements	array	Mapa elementos a ser criado no mapa.
urls	array	Mapa URLs a ser criado no mapa.
users	array	Mapeie os compartimentos de usuário a serem criados no mapa.
userGroups	array	Map user group shares to be created on the map.
shapes	array	Mapa shapes a ser criado no mapa.
lines	array	Mapa linhas a ser criado no mapa.

Note:

Para criar links de mapa, você precisará definir um elemento de mapa `selementid` para um valor arbitrário e, em seguida, usar esse valor para fazer referência a esse elemento nas propriedades `selementid1` ou `selementid2` dos links. Quando o elemento for criado, este valor será substituído pelo ID correto gerado pelo Zabbix. [Veja exemplo](#).

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas criados na propriedade `sysmapids`. A ordem dos IDs retornados corresponde à ordem dos mapas passados.

Exemplos

Criar um mapa vazio

Criar um mapa sem elementos.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Map",
    "width": 600,
    "height": 600
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "8"
    ]
  },
  "id": 1
}
```

Criar um mapa de host

Crie um mapa com dois elementos hospedeiros e um link entre eles. Observe o uso de valores temporários "selementid1" e "selementid2" no objeto de link do mapa para fazer referência aos elementos do mapa.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Host map",
    "width": 600,
    "height": 600,
    "selements": [
      {
        "selementid": "1",
        "elements": [
          {"hostid": "1033"}
        ],
        "elementtype": 0,
        "iconid_off": "2"
      },
      {
        "selementid": "2",
        "elements": [
          {"hostid": "1037"}
        ],
        "elementtype": 0,
        "iconid_off": "2"
      }
    ],
    "links": [
      {
        "selementid1": "1",
        "selementid2": "2"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
}
```

```
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "9"
    ]
  },
  "id": 1
}
```

Criar um trigger mapa

Crie um mapa com elemento de trigger, que contém duas triggers.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Trigger map",
    "width": 600,
    "height": 600,
    "selements": [
      {
        "elements": [
          {"triggerid": "12345"},
          {"triggerid": "67890"}
        ],
        "elementtype": 2,
        "iconid_off": "2"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "10"
    ]
  },
  "id": 1
}
```

Compartilhamento de mapa

Crie um mapa com dois tipos de compartilhamento (usuário e grupo de usuários).

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Map sharing",
    "width": 600,
    "height": 600,
```

```

    "users": [
      {
        "userid": "4",
        "permission": "3"
      }
    ],
    "userGroups": [
      {
        "usrgrpid": "7",
        "permission": "2"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "9"
    ]
  },
  "id": 1
}

```

Formas do mapa

Criar um mapa com o título do nome do mapa.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Host map",
    "width": 600,
    "height": 600,
    "shapes": [
      {
        "type": 0,
        "x": 0,
        "y": 0,
        "width": 600,
        "height": 11,
        "text": "{MAP.NAME}"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "10"
    ]
  },
  "id": 1
}

```



```
    "id": 1
}
```

Linhas do mapa

Crie uma linha de mapa..

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.create",
  "params": {
    "name": "Map API lines",
    "width": 500,
    "height": 500,
    "lines": [
      {
        "x1": 30,
        "y1": 10,
        "x2": 100,
        "y2": 50,
        "line_type": 1,
        "line_width": 10,
        "line_color": "009900"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "11"
    ]
  },
  "id": 1
}
```

Veja também

- [Map element](#)
- [Map link](#)
- [Map URL](#)
- [Map user](#)
- [Map user group](#)
- [Map shape](#)
- [Map line](#)

Fonte

CMap::create() in ui/include/classes/api/services/CMap.php.

map.delete

Descrição

object map.delete(array mapIds)

Este método permite excluir mapas.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos mapas a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas excluídos na propriedade 'sysmapids'.

Exemplos**Excluir vários mapas**

Excluir dois mapas.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.delete",
  "params": [
    "12",
    "34"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "sysmapids": [
      "12",
      "34"
    ]
  },
  "id": 1
}
```

Fonte

CMap::delete() in ui/include/classes/api/services/CMap.php.

map.get**Descrição**

integer/array map.get(object parameters)

O método permite recuperar mapas de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
sysmapids	string/array	Retorna apenas mapas com os IDs fornecidos.

Parâmetro	Tipo	Descrição
userids	string/array	Retorna apenas mapas que pertencem aos IDs de usuário fornecidos.
expandUrls	flag	Adiciona URLs de mapas globais aos elementos de mapa correspondentes e expande macros em todos os URLs de elementos de mapa.
selectIconMap	query	Retorna uma propriedade iconmap com o mapa de ícones usado no mapa.
selectLinks	query	Retorna uma propriedade links com os links do mapa entre os elementos.
selectSelements	query	Retorna uma propriedade selements com os elementos do mapa.
selectUrls	query	Retorna uma propriedade urls com os URLs do mapa.
selectUsers	query	Retorna uma propriedade users com usuários com os quais o mapa é compartilhado.
selectUserGroups	query	Retorna uma propriedade userGroups com grupos de usuários com os quais o mapa é compartilhado.
selectShapes	query	Retorna uma propriedade shapes com as formas do mapa.
selectLines	query	Retorna uma propriedade linhas com as linhas do mapa.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: name, width e height. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar um mapa

Recupere todos os dados sobre o mapa "3".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.get",
  "params": {
    "output": "extend",
    "selectSelements": "extend",
    "selectLinks": "extend",
    "selectUsers": "extend",
    "selectUserGroups": "extend",
    "selectShapes": "extend",
    "selectLines": "extend",
    "sysmapids": "3"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
```

```
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "selements": [
        {
          "selementid": "10",
          "sysmapid": "3",
          "elementtype": "4",
          "evaltype": "0",
          "iconid_off": "1",
          "iconid_on": "0",
          "label": "Zabbix server",
          "label_location": "3",
          "x": "11",
          "y": "141",
          "iconid_disabled": "0",
          "iconid_maintenance": "0",
          "elementsubtype": "0",
          "areatype": "0",
          "width": "200",
          "height": "200",
          "tags": [
            {
              "tag": "service",
              "value": "mysqld",
              "operator": "0"
            }
          ],
          "viewtype": "0",
          "use_iconmap": "1",
          "urls": [],
          "elements": []
        },
        {
          "selementid": "11",
          "sysmapid": "3",
          "elementtype": "4",
          "evaltype": "0",
          "iconid_off": "1",
          "iconid_on": "0",
          "label": "Web server",
          "label_location": "3",
          "x": "211",
          "y": "191",
          "iconid_disabled": "0",
          "iconid_maintenance": "0",
          "elementsubtype": "0",
          "areatype": "0",
          "width": "200",
          "height": "200",
          "viewtype": "0",
          "use_iconmap": "1",
          "tags": [],
          "urls": [],
          "elements": []
        }
      ]
    }
  ]
}
```

```

        "selementid": "12",
        "sysmapid": "3",
        "elementtype": "0",
        "evaltype": "0",
        "iconid_off": "185",
        "iconid_on": "0",
        "label": "{HOST.NAME}\\r\\n{HOST.CONN}",
        "label_location": "0",
        "x": "111",
        "y": "61",
        "iconid_disabled": "0",
        "iconid_maintenance": "0",
        "elementsubtype": "0",
        "areatype": "0",
        "width": "200",
        "height": "200",
        "viewtype": "0",
        "use_iconmap": "0",
        "tags": [],
        "urls": [],
        "elements": [
            {
                "hostid": "10084"
            }
        ]
    },
    ],
    "links": [
        {
            "linkid": "23",
            "sysmapid": "3",
            "selementid1": "10",
            "selementid2": "11",
            "drawtype": "0",
            "color": "00CC00",
            "label": "",
            "linktriggers": []
        }
    ],
    "users": [
        {
            "sysmapuserid": "1",
            "userid": "2",
            "permission": "2"
        }
    ],
    "userGroups": [
        {
            "sysmapusrgrpid": "1",
            "usrgrpid": "7",
            "permission": "2"
        }
    ],
    "shapes": [
        {
            "sysmap_shapeid": "1",
            "type": "0",
            "x": "0",
            "y": "0",
            "width": "680",
            "height": "15",
            "text": "{MAP.NAME}",

```

```

        "font": "9",
        "font_size": "11",
        "font_color": "000000",
        "text_halign": "0",
        "text_valign": "0",
        "border_type": "0",
        "border_width": "0",
        "border_color": "000000",
        "background_color": "",
        "zindex": "0"
    }
],
"lines": [
    {
        "sysmap_shapeid": "2",
        "x1": 30,
        "y1": 10,
        "x2": 100,
        "y2": 50,
        "line_type": 1,
        "line_width": 10,
        "line_color": "009900",
        "zindex": "1"
    }
],
"sysmapid": "3",
"name": "Local network",
"width": "400",
"height": "400",
"backgroundid": "0",
"label_type": "2",
"label_location": "3",
"highlight": "1",
"expandproblem": "1",
"markelements": "0",
"show_unack": "0",
"grid_size": "50",
"grid_show": "1",
"grid_align": "1",
"label_format": "0",
"label_type_host": "2",
"label_type_hostgroup": "2",
"label_type_trigger": "2",
"label_type_map": "2",
"label_type_image": "2",
"label_string_host": "",
"label_string_hostgroup": "",
"label_string_trigger": "",
"label_string_map": "",
"label_string_image": "",
"iconmapid": "0",
"expand_macros": "0",
"severity_min": "0",
"userid": "1",
"private": "1",
"show_suppressed": "1"
}
],
"id": 1
}

```

Veja também

- [Icon map](#)
- [Map element](#)
- [Map link](#)
- [Map URL](#)
- [Map user](#)
- [Map user group](#)
- [Map shapes](#)
- [Map lines](#)

Fonte

CMap::get() in ui/include/classes/api/services/CMap.php.

map.update

Descrição

`object map.update(object/array maps)`

Este método permite atualizar os mapas existentes.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do mapa a serem atualizadas.

A propriedade `mapid` deve ser definida para cada mapa, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades padrão do mapa](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
links	array	Mapeie links para substituir os links existentes.
selements	array	Mapeie elements para substituir os elementos existentes.
urls	array	Mapeie URLs para substituir os URLs existentes.
users	array	Mapeie os compartilhamentos de user para substituir os elementos existentes.
userGroups	array	Mapeie os compartilhamentos de grupo de usuários para substituir os elementos existentes.
shapes	array	Mapeie formas para substituir as formas existentes.
lines	array	Mapeie linhas para substituir as linhas existentes.

Note:

Para criar links de mapa entre novos elementos de mapa, você precisará definir o `selementid` de um elemento para um valor arbitrário e, em seguida, usar esse valor para fazer referência a esse elemento nas propriedades `selementid1` ou `selementid2` dos links. Quando o elemento for criado, este valor será substituído pelo ID correto gerado pelo Zabbix. [Veja exemplo para map.create.](#)

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas atualizados na propriedade `sysmapids`.

Exemplos

Redimensionar um mapa

Altere o tamanho do mapa para 1200x1200 pixels..

Request:

```
{
  "jsonrpc": "2.0",
  "method": "map.update",
```

```

    "params": {
        "sysmapid": "8",
        "width": 1200,
        "height": 1200
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "8"
        ]
    },
    "id": 1
}

```

Alterar proprietário do mapa

Disponível apenas para administradores e super admins.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "map.update",
    "params": {
        "sysmapid": "9",
        "userid": "1"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 2
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "sysmapids": [
            "9"
        ]
    },
    "id": 2
}

```

Veja também

- [Map element](#)
- [Map link](#)
- [Map URL](#)
- [Map user](#)
- [Map user group](#)
- [Map shapes](#)
- [Map lines](#)

Fonte

CMap::update() in ui/include/classes/api/services/CMap.php.

Mapa de ícones

Esta classe foi projetada para trabalhar com mapas de ícones.

Referências de objetos:

- [Mapa de ícones](#)
- [Mapeamento de ícones](#)

Métodos disponíveis:

- [iconmap.create](#) - criar novo mapas de ícones
- [iconmap.delete](#) - ícone de exclusão mapas
- [iconmap.get](#) - ícone de recuperação mapas
- [iconmap.update](#) - ícone de atualização mapas

> Objeto de mapa de ícones

Os objetos a seguir estão diretamente relacionados à API `iconmap`.

Mapa de ícones

O objeto mapa de ícones contém as propriedades abaixo.

Propriedade	Tipo	Descrição
<code>iconmapid</code>	string	(somente leitura) ID do mapa de ícones.
<code>default_iconid</code> (obrigatório)	string	ID do ícone padrão.
<code>name</code> (obrigatório)	string	Nome do mapa de ícones.

Observe que para alguns métodos (`update`, `delete`) a combinação de parâmetros obrigatórios/opcionais é diferente.

Mapeamento de ícones

O objeto de mapeamento de ícones define um ícone específico a ser usado para hosts com um determinado valor de campo de inventário. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
<code>iconmappingid</code>	string	(somente leitura) ID do mapa de ícones.
<code>iconid</code> (obrigatório)	string	ID do ícone usado pelo mapeamento de ícones.
<code>expressão</code> (obrigatório)	string	Expressão para corresponder ao campo de inventário.
<code>inventário_link</code> (obrigatório)	inteiro	ID do campo de inventário do host.
		Consulte o objeto de inventário do host para obter uma lista de campos de inventário com suporte.
<code>iconmapid</code>	string	(somente leitura) ID do mapa de ícones ao qual o mapeamento de ícones pertence.
<code>sortorder</code>	integer	(somente leitura) Posição do mapeamento de ícones no mapa de ícones.

`iconmap.create`

Descrição

```
object iconmap.create(object/array iconMaps)
```

Este método permite criar novos mapas de ícones.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object/array) Mapas de ícones para criar.

Além das **propriedades padrão do mapa de ícones**, o O método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| **mapeamentos**
(obrigatório)|array|Mapeamentos de ícones a serem criados para o mapa de ícones.]

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas de ícones criados sob a propriedade `iconmapids`. A ordem dos IDs retornados corresponde a ordem dos mapas de ícones passados.

Exemplos

Crie um mapa de ícones

Crie um mapa de ícones para exibir hosts de diferentes tipos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "iconmap.create",
  "parâmetros": {
    "name": "Tipo de ícones",
    "default_iconid": "2",
    "mapeamentos": [
      {
        "link_inventário": 1,
        "expressão": "servidor",
        "ícone": "3"
      },
      {
        "link_inventário": 1,
        "expressão": "interruptor",
        "iconide": "4"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "iconmapids": [
      "2"
    ]
  },
  "id": 1
}
```

Veja também

- **Mapeamento de ícones**

Fonte

`ClconMap::create()` em `ui/include/classes/api/services/ClconMap.php`.

iconmap.delete

Descrição

`object iconmap.delete(array iconMapIds)`

Este método permite excluir mapas de ícones.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(array) IDs dos mapas de ícones a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas de ícones excluídos sob a propriedade `iconmapids`.

Exemplos

Excluir vários mapas de ícones

Exclua dois mapas de ícones.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "iconmap.delete",
  "parâmetros": [
    "2",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "iconmapids": [
      "2",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

`ClconMap::delete()` em `ui/include/classes/api/services/ClconMap.php`.

iconmap.get

Descrição

`integer/array iconmap.get(object parameters)`

O método permite recuperar mapas de ícones de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

[Parâmetro Tipo Descrição]	[iconmapids string/array Retorna apenas mapas de ícones com os IDs fornecidos.]	[sysmapids string/array Retorna apenas mapas de ícones que são usados nos mapas fornecidos.]

|selectMappings|query|Retorne uma propriedade **mappings** com os mapeamentos de ícone usados.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: iconmapid e name.| |countOutput|boolean|Estes parâmetros sendo comuns para todos os métodos get são descritos em detalhes no **comentário de referência**.| |editável|booleano|^| |excludePesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar um mapa de ícones

Recupere todos os dados sobre o mapa de ícones "3".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "iconmap.get",
  "parâmetros": {
    "iconmapids": "3",
    "saída": "estender",
    "selectMappings": "estender"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "mapeamentos": [
        {
          "iconmappingid": "3",
          "iconmapid": "3",
          "iconid": "6",
          "link_inventário": "1",
          "expressão": "servidor",
          "ordem de classificação": "0"
        },
        {
          "iconmappingid": "4",
          "iconmapid": "3",
          "iconid": "10",
          "link_inventário": "1",
          "expressão": "interruptor",
          "ordem de classificação": "1"
        }
      ],
      "iconmapid": "3",
      "name": "Ícones de tipo de host",
      "default_iconid": "2"
    }
  ],
  "id": 1
}
```

Veja também

- **Mapeamento de ícones**

Fonte

ClconMap::get() em ui/include/classes/api/services/ClconMap.php.

iconmap.update

Descrição

object iconmap.update(object/array iconMaps)

Este método permite atualizar os mapas de ícones existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do mapa de ícones a serem atualizadas.

A propriedade iconmapid deve ser definida para cada mapa de ícones, todos os outros propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas os outros permanecerão inalterados.

Além das [propriedades padrão do mapa de ícones](#), o método aceita os seguintes parâmetros.

|Parâmetro|Tipo|Descrição| |-----|-----| |mappings|array|Mapeamentos de ícones para substituir os mapeamentos de ícones existentes.|

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas de ícones atualizados sob a propriedade iconmapids.

Exemplos

Renomear mapa de ícones

Renomeie um mapa de ícones para "Ícones do SO".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "iconmap.update",
  "parâmetros": {
    "iconmapid": "1",
    "name": "Ícones do SO"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "iconmapids": [
      "1"
    ]
  },
  "id": 1
}
```

Veja também

- [Mapeamento de ícones](#)

Fonte

ClconMap::update() em ui/include/classes/api/services/ClconMap.php.

Nó de alta disponibilidade

Esta classe foi projetada para trabalhar com nós de servidor que fazem parte de um High cluster de disponibilidade ou uma instância de servidor independente.

Referências de objetos:

- [Nodo de alta disponibilidade](#)

Métodos disponíveis:

- [hanode.get](#) - recuperando nós

> Objeto de nó de alta disponibilidade

O objeto a seguir está relacionado à operação de um cluster de alta disponibilidade dos servidores Zabbix.

Nó de alta disponibilidade

Note:

Os nós são criados pelo servidor Zabbix e não podem ser modificados por meio da API.

O objeto de nó de alta disponibilidade tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----| |ha_nodeid|string|ID do nó.| |name|string|Nome atribuído ao nó, usando a entrada de configuração HANodeName de zabbix_server.conf. Vazio para um servidor em execução no modo autônomo.| |address|string|IP ou nome DNS de onde o nó se conecta.| |port|integer|Porta na qual o nó está sendo executado.| |lastaccess|integer|Heartbeat time, t.i. hora da última atualização do nó. Carimbo de hora UTC.| |status|integer|Estado do nó.

Valores possíveis:
0 - espera;
1 - parado manualmente;
2 - indisponível;
3 - ativo.|

hanode.get

Descrição

integer/array hanode.get(object parameters)

O método permite recuperar uma lista de nós de cluster de alta disponibilidade de acordo com os parâmetros fornecidos.

Note:

Este método está disponível apenas para os tipos de usuário Super admin. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----| |ha_nodeids|string/array|Retorna apenas nós com os IDs de nó fornecidos.| |filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.

Permite filtrar pelas propriedades do nó: name, address, status.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: name, lastaccess, status.| |countOutput|flag|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no [comentário de referência](#).| |limite|integer|^| |saída|consulta|^| |preservekeys|boolean|^| |ordem de classificação|cadeia/matriz|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Obtenha uma lista de nós ordenados por status

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hanode.get",
  "parâmetros": {
    "preservekeys": verdadeiro,
    "sortfield": "status",
    "ordem de classificação": "DESC"
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "ckuo7i1nw000h0sajj3l3hh8u": {
      "ha_nodeid": "ckuo7i1nw000h0sajj3l3hh8u",
      "name": "nó ativo",
      "endereço": "192.168.1.13",
      "porta": "10051",
      "lastaccess": "1635335704",
      "estado": "3"
    },
    "ckuo7i1nw000e0sajwfttc1mp": {
      "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
      "nome": "nó6",
      "endereço": "192.168.1.10",
      "porta": "10053",
      "lastaccess": "1635332902",
      "estado": "2"
    },
    "ckuo7i1nv000c0sajz85xcrtt": {
      "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
      "nome": "nó4",
      "endereço": "192.168.1.8",
      "porta": "10052",
      "lastaccess": "1635334214",
      "estado": "1"
    },
    "ckuo7i1nv000a0saj1fcdkeu4": {
      "ha_nodeid": "ckuo7i1nv000a0saj1fcdkeu4",
      "nome": "nó2",
      "endereço": "192.168.1.6",
      "porta": "10051",
      "lastaccess": "1635335705",
      "estado": "0"
    }
  },
  "id": 1
}
```

Obtenha uma lista de nós específicos por seus IDs

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hanode.get",
  "parâmetros": {
    "ha_nodeids": ["ckuo7i1nw000e0sajwfttc1mp", "ckuo7i1nv000c0sajz85xcrtt"]
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

```
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
      "nome": "nô4",
      "endereço": "192.168.1.8",
      "porta": "10052",
      "lastaccess": "1635334214",
      "estado": "1"
    },
    {
      "ha_nodeid": "ckuo7i1nw000e0sajwfttc1mp",
      "nome": "nô6",
      "endereço": "192.168.1.10",
      "porta": "10053",
      "lastaccess": "1635332902",
      "estado": "2"
    }
  ],
  "id": 1
}
```

Obter uma lista de nós parados

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hanode.get",
  "parâmetros": {
    "output": ["ha_nodeid", "endereço", "porta"],
    "filtro": {
      "estado": 1
    }
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "ha_nodeid": "ckuo7i1nw000g0sajjsjre7e3",
      "endereço": "192.168.1.12",
      "porta": "10051"
    },
    {
      "ha_nodeid": "ckuo7i1nv000c0sajz85xcrtt",
      "endereço": "192.168.1.8",
      "porta": "10052"
    },
    {
      "ha_nodeid": "ckuo7i1nv000d0sajd95y1b6x",
      "endereço": "192.168.1.9",
      "porta": "10053"
    }
  ],
}
```



```
    "id": 1
}
```

Obtenha uma contagem de nós em espera

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hanode.get",
  "parâmetros": {
    "countOutput": verdadeiro,
    "filtro": {
      "estado": 0
    }
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": "3",
  "id": 1
}
```

Verifique o status dos nós em endereços IP específicos

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hanode.get",
  "parâmetros": {
    "saída": ["nome", "status"],
    "filtro": {
      "endereço": ["192.168.1.7", "192.168.1.13"]
    }
  },
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "nome": "nó3",
      "estado": "0"
    },
    {
      "name": "nó ativo",
      "estado": "3"
    }
  ],
  "id": 1
}
```

Fonte

CHaNode::get() em ui/include/classes/api/services/CHaNode.php.

Painel

Esta classe foi projetada para trabalhar com painéis.

Referências de objetos:

- [Dashboard](#)
- [Painel página](#)
- [Painel widget](#)
- [Widget do painel campo](#)
- [Painel usuário](#)
- [Usuário do painel group](#)

Métodos disponíveis:

- [dashboard.create](#) - criando novos painéis
- [dashboard.delete](#) - excluindo painéis
- [dashboard.get](#) - recuperando painéis
- [dashboard.update](#) - atualizando painéis

> Objeto do painel

Os objetos a seguir estão diretamente relacionados à API dashboard.

Painel

O objeto de painel tem as seguintes propriedades.

Propriedade	Tipo	Descrição
dashboardid	string	(readonly) ID do painel.
name (obrigatório)	string	Nome do painel.
userid	string	ID do usuário do proprietário do painel.
private	inteiro	Tipo de compartilhamento do painel. Valores possíveis: 0 - painel público; 1 - (padrão) painel privado.
display_period	inteiro	Período de exibição da página padrão (em segundos). Valores possíveis: 10, 30, 60, 120, 600, 1800, 3600.
auto_start	integer	Padrão: 30. Iniciar apresentação de slides automaticamente. Valores possíveis: 0 - não iniciar apresentação de slides automaticamente; 1 - (padrão) iniciar apresentação de slides automaticamente.

Observe que para alguns métodos (update, delete) a combinação de parâmetros obrigatórios/opcionais é diferente.

Página do painel

O objeto de página do painel tem as seguintes propriedades.

Propriedade	Tipo	Descrição
dashboard_pageid	string	(readonly) ID da página do painel.
name	string	Nome da página do painel. Padrão: string vazia.

Propriedade	Tipo	Descrição
display_period	inteiro	Período de exibição da página do painel (em segundos). Valores possíveis: 0, 10, 30, 60, 120, 600, 1800, 3600.
widgets	array	Padrão: 0 (usará o período de exibição de página padrão). Array dos objetos widget do painel .

Widget do painel

O objeto widget do painel tem as seguintes propriedades.

Propriedade	Tipo	Descrição
widgetid	string	(readonly) ID do widget do painel.
type (obrigatório)	string	Tipo do widget do painel. Valores possíveis: actionlog - Log de ações; clock - Relógio; (preterido) dataover - Visão geral dos dados; discovery - Status da descoberta; favgraphs - Gráficos favoritos; favmaps - Mapas favoritos; graph - Gráfico (clássico); graphprototype - Protótipo do gráfico; hostavail - Disponibilidade do host; item - Valor do item; map - Mapa; navtree - Árvore de navegação do mapa; plaintext - Texto puro; problemhosts - Hosts com problema; problems - Problemas; problemsbysv - Problemas por gravidade; slareport - Relatório SLA; svggraph - Gráfico; systeminfo - Informações do sistema; tophosts - Principais hosts; trigover - Visão geral do acionador; url - URL; web - Monitoramento da web.
name	string	Nome do widget personalizado.
x	inteiro	Uma posição horizontal do lado esquerdo do painel.
y	inteiro	Os valores válidos variam de 0 a 23. Uma posição vertical da parte superior do painel.
width	integer	Os valores válidos variam de 0 a 62. A largura do widget.
altura	inteiro	Os valores válidos variam de 1 a 24. A altura do widget.
view_mode	integer	Os valores válidos variam de 2 a 32. O modo de visualização de widget. Valores possíveis: 0 - (padrão) visualização de widget padrão; 1 - com cabeçalho oculto;

Propriedade	Tipo	Descrição
fields	array	Array dos objetos campo do widget do painel . Obrigatório para widgets específicos; veja widgets individuais em Campos de widget do painel .

Objeto de campo do widget de painel

O objeto de campo do widget de painel possui as seguintes propriedades.

Propriedade	Type	Descrição
type (obrigatório)	inteiro	Tipo do campo de widget. Valores possíveis: 0 - Inteiro; 1 - String; 2 - Grupo de hosts; 3 - Host; 4 - Item; 5 - Protótipo item; 6 - Gráfico; 7 - Protótipo gráfico; 8 - Mapa; 9 - Serviço; 10 - SLA.
name (obrigatório)	string	Nome do campo do widget.
value (obrigatório)	variado	Valores possíveis: consulte manual Dashboard widget fields . Valor do campo do widget dependendo do tipo. Valores possíveis: consulte manual Dashboard widget fields .

Grupo de usuários do Dashboard

Lista de permissões do painel com base em grupos de usuários. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
usrgrpId (required)	string	ID do grupo de usuários.
permission (required)	integer	Tipo de nível de permissão. Valores possíveis: 2 - somente leitura; 3 - leitura-escrita;

Usuário do Dashboard

Lista de permissões do painel com base em usuários. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
userId (required)	string	ID do usuário.
permission (required)	integer	Tipo de nível de permissão. Valores possíveis: 2 - somente leitura; 3 - leitura-escrita;

Campos de widgets no painel de controle

Esta página contém links de navegação para os parâmetros do widget do painel de controle e os possíveis valores de propriedade para os respectivos objetos `dashboard.widget.field`.

Para ver os parâmetros e os valores de propriedade para cada widget, vá para as páginas individuais de widgets:

- [Action log](#)
- [Clock](#)
- [Discovery status](#)
- [Favorite graphs](#)
- [Favorite maps](#)
- [Geomap](#)
- [Graph](#)
- [Graph \(classic\)](#)
- [Graph prototype](#)
- [Host availability](#)
- [Item value](#)
- [Map](#)
- [Map navigation tree](#)
- [Plain text](#)
- [Problem hosts](#)
- [Problems](#)
- [SLA report](#)
- [System information](#)
- [Problems by severity](#)
- [Top hosts](#)
- [Trigger overview](#)
- [URL](#)
- [Web monitoring](#)

Widgets descontinuados:

- [Data overview](#)

Attention:

Widgets descontinuados serão removidos na próxima versão.

1 Action log

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the `Action log` widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the Action log widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - (default) 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - 15 minutes.

Parameter	type	name	value
Sort entries by	0	sort_triggers	3 - Time (ascending); 4 - (default) Time (descending); 5 - Type (ascending); 6 - Type (descending); 7 - Status (ascending); 8 - Status (descending); 11 - Recipient (ascending); 12 - Recipient (descending).
Show lines	0	show_lines	Valid values range from 1-100. Default: 25.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Action log widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring an Action log widget

Configure an Action log widget that displays 10 entries of action operation details, sorted by time (in ascending order).

Request:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "actionlog",
            "name": "Action log",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "show_lines",
                "value": 10
              },
              {
                "type": 0,
                "name": "sort_triggers",
                "value": 3
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
    {
      "usrgrp_id": 7,
      "permission": 2
    }
  ],
}
```

```

    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ],
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
  }
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

See also

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

2 Clock

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget de painel respectivos permitem configurar o widget **Clock** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos de Widget fields não são validadas durante a criação ou atualização de um painel. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Clock, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Clock.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - (padrão) 15 minutos.
Tipo de hora	0	time_type	0 - (padrão) Hora local; 1 - Hora do servidor; 2 - Hora do host.

Os seguintes parâmetros são suportados se o Tipo de hora for definido como "Hora do host".

Parâmetro	type	nome	valor
Item (obrigatório)	4	iditem	Item ID do item.

Exemplos

Os seguintes exemplos visam apenas descrever a configuração do campo objetos de dashboard widget para o o widget relógio (clock). Para mais informações sobre configuração de dashboard, consulte [dashboard.create](#).

Configurando um widget Clock

Configure um widget Clock que exiba a hora do servidor Zabbix.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "clock",
            "name": "Clock",
            "x": 0,
            "y": 0,
            "width": 4,
            "height": 3,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "time_type",
                "value": 1
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
        "usrgrp_id": 7,
        "permission": 2
      }
    ],
    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
```



```

    "result": {
      "dashboardids": [
        "3"
      ]
    },
    "id": 1
  }

```

Consulte também

- [Campo widget do dashboard](#)
- [dashboard.create](#)
- [dashboard.update](#)

3 Visão geral dos dados

Attention:

Este widget foi descontinuado e será removido na próxima grande atualização. Considere utilizar os widgets [Top hosts](#).

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget permitem configurar o [Data overview](#) nos métodos `dashboard.create` e `dashboard.update`.

Parâmetros

Os parâmetros seguintes são suportados pelo Widget Data overview.

Parameter	type	nome	valor
Refresh interval	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Host groups	2	groupids	Host group ID. Note: Para configurar vários grupos de host, crie um objeto de campo de widget no painel de cada grupo de host.
Hosts	3	hostids	Host ID. Note: Para configurar vários hosts, crie um objeto de campo de widget no painel de cada host. Para vários hosts, o parâmetro Host groups deve estar ou não configurado, ou configurado com pelo menos um grupo de hosts ao qual os hosts configurados pertencem.

Parameter	type	nome	valor
Tags (the number in the property name (e.g. tags.tag.0) references tag order in the tag evaluation list)			
Evaluation type	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Tag name	1	tags.tag.0	Qualquer valor de string.
Operator	0	tags.operator.0	Parâmetro Tag name é necessário ao configurar Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Tag value	1	tags.value.0	Parâmetro Operator é necessário ao configurar Tags. Qualquer valor de string.
Show suppressed problems	0	show_suppressed	Parâmetro Tag value necessário ao configurar Tags. 0 - (padrão) Desabilitado; 1 - Habiitado.
Hosts location	0	style	0 - (padrão) Esquerda; 1 - Topo.

Exemplos

Os seguintes exemplos visam apenas a descrever a configuração dos objetos de campo de widget do painel para widget Data overview. Para mais informações sobre a configuração de um painel, veja [dashboard.create](#).

Configurar uma Data overview widget

Configure uma Data overview widget que exibe dados para host "10084" e apenas para itens nos quais a tag com nome "component" contém valor "cpu". Além disso, exiba os dados com hosts localizados no topo.

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "dataover",
            "name": "Data overview",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 3,
                "name": "hostids",
                "value": 10084
              },
              {
                "type": 1,
                "name": "tags.tag.0",
                "value": "component"
              },
              {
                "type": 0,
                "name": "tags.operator.0",
                "value": 0
              },
              {
                "type": 1,
                "name": "tags.value.0",
                "value": "cpu"
              },
              {
                "type": 0,
                "name": "style",
                "value": 1
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
        "usrgrpid": 7,
        "permission": 2
      }
    ],
    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ]
  }
},

```

```

    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
  }

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Veja também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

4 Discovery status

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the **Discovery status** widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the Discovery status widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - (default) 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - 15 minutes.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Discovery status widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring Discovery status widget

Configure a Discovery status widget with the refresh interval set to 15 minutes.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "discovery",

```

```

        "name": "Discovery status",
        "x": 0,
        "y": 0,
        "width": 6,
        "height": 3,
        "view_mode": 0,
        "fields": [
            {
                "type": 0,
                "name": "rf_rate",
                "value": 900
            }
        ]
    },
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

See also

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

5 Favorite graphs

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the **Favorite graphs** widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the Favorite graphs widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - (default) 15 minutes.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Favorite graphs widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring a Favorite graphs widget

Configure a Favorite graphs widget with the refresh interval set to 10 minutes.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "favgraphs",
            "name": "Favorite graphs",
            "x": 0,
            "y": 0,
            "width": 4,
            "height": 3,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "rf_rate",
                "value": 600
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
        "usrgrpId": 7,
        "permission": 2
      }
    ],
    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
}
```

```
    "id": 1
  }
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

See also

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

6 Favorite maps

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the **Favorite maps** widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the Favorite maps widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - (default) 15 minutes.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Favorite maps widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring a Favorite maps widget

Configure a Favorite maps widget with the refresh interval set to 10 minutes.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "favmaps",
            "name": "Favorite maps",

```

```

        "x": 0,
        "y": 0,
        "width": 4,
        "height": 3,
        "view_mode": 0,
        "fields": [
            {
                "type": 0,
                "name": "rf_rate",
                "value": 600
            }
        ]
    },
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

See also

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

7 Geomap

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo do widget **Geomap** permitem configurar o widget em métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do widget fields não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os **built-in widgets** e criem **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widge Geomap, consulte os requisitos de parâmetros descritos nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Geomap.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de host	2	groupids	Host group ID.
Hosts	3	hostids	Host ID.
<p>Observação: Para configurar vários grupos de hosts, crie um objeto de campo do widget do dashboard para cada grupo de hosts.</p> <p>Note: Para configurar vários hosts, crie um objeto de campo do widget do painel para cada host. Para vários hosts, o parâmetro grupos de host deve estar não configurado ou configurado com pelo menos um grupo de hosts ao qual os hosts configurados pertencem.</p>			
Tags (o número no nome da propriedade (por exemplo tags.tag.0) refere-se à ordem de avaliação da tag na lista de avaliação de tags)			
Tipo avaliação	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.

Parâmetro	type	nome	valor
Nome da tag	1	tags.tag.0	Qualquer valor de string.
Operator	0	tags.operator.0	Parâmetro Nome da tag necessário se estiver configurando Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Valor da tag	1	tags.value.0	Parâmetro Operador necessário se estiver configurando Tags. Qualquer valor de string.
Visualização inicial	1	default_view	Parâmetro Valor da tag required if configuring Tags. latitude, longitude, nível de zoom separados por vírgula (opcional, valores válidos variam de 0-30). Exemplo: 40.6892494,-74.0466891,10.

Exemplos

Os seguintes exemplos visam apenas descrever a configuração dos objetos de campo do widget do painel para o widget Geomap. Para mais informações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget Geomap

Para configurar um widget Geomap que exiba hosts dos grupos de hosts "2" e "22" com base na seguinte configuração de tags: Tag com o nome "component" contém o valor "node", ou Tag com o nome "location" é igual ao valor "New York". Além disso, defina a visualização inicial do mapa para as coordenadas "40.6892494" (latitude), "-74.0466891" (longitude) com o nível de zoom "10", você pode usar o seguinte:

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "geomap",
            "name": "Geomap",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 2,
                "name": "groupids",
                "value": 22
              },
              {
                "type": 2,
                "name": "groupids",
                "value": 2
              },
              {
                "type": 1,
```

```

        "name": "default_view",
        "value": "40.6892494,-74.0466891,10"
    },
    {
        "type": 0,
        "name": "evaltype",
        "value": 2
    },
    {
        "type": 1,
        "name": "tags.tag.0",
        "value": "component"
    },
    {
        "type": 0,
        "name": "tags.operator.0",
        "value": 0
    },
    {
        "type": 1,
        "name": "tags.value.0",
        "value": "node"
    },
    {
        "type": 1,
        "name": "tags.tag.1",
        "value": "location"
    },
    {
        "type": 0,
        "name": "tags.operator.1",
        "value": 1
    },
    {
        "type": 1,
        "name": "tags.value.1",
        "value": "New York"
    }
}
]
}
],
"userGroups": [
    {
        "usrgrpId": 7,
        "permission": 2
    }
],
"users": [
    {
        "userid": 1,
        "permission": 3
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

Consulte também

- [Campo widget do dashboard](#)
- [dashboard.create](#)
- [dashboard.update](#)

8 Gráfico

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo do widget de dashboard permitem configurar o widget **Graph** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos `Widget fields properties` não são validadas durante a criação ou atualização de um dashboard. Isso permite aos usuários modificar **built-in widgets** e criar **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget **Graph**, consulte os requisitos de parâmetro descritos nas tabelas abaixo.

Parâmetros

The following parameters are supported for the Graph widget.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.

Conjunto de dados

Os seguintes parâmetros são suportados para configurar um Conjunto de dados.

Note:

O primeiro número no nome da propriedade (por exemplo, `ds.hosts.0.0`, `ds.items.0.0`) representa o conjunto de dados específico, enquanto o segundo número, se presente, representa o host ou item configurado.

Parâmetro	type	nome	valor
Padrão de host (obrigatório)	1	ds.hosts.0.0	Nome ou padrão de Host (por exemplo <code>Zabbix*</code>).
Padrão e item (obrigatório)	1	ds.items.0.0	Nome ou padrão de Item (por exemplo <code>*: Number of processed values per second</code>).
Cor	1	ds.color.0	Código de cor hexadecimal (por exemplo, <code>FF0000</code>). Padrão: <code>FF465C</code> .

Parâmetro	type	nome	valor
Desenhar	0	ds.type.0	0 - (padrão) Linha; 1 - Pontos; 2 - Escada; 3 - Barra.
Largura	0	ds.width.0	Valores válidos variam entre 1-10. Padrão: 1. Parâmetro Largura não disponível se o Desenhar estiver configurado como "Pontos" ou "Barra".
Tamanho do ponto	0	ds.pointsize.0	Valores válidos variam entre 1-10. Padrão: 3. Parâmetro Tamanho do ponto não disponível se Desenhar estiver definido como "Linha", "Escada" or "Barra".
Transparência	0	ds.transparency.0	Valores válidos variam entre 1-10. Padrão: 5.
Preenchimento	0	ds.fill.0	Valores válidos variam entre 1-10. Padrão: 3. Parâmetro Preenchimento não disponível se Desenhar estiver configurado como "Pontos" ou "Barra".
Missing data	0	ds.missingdatafunc.0	0 - (padrão) Nenhum; 1 - Conectado; 2 - Tratar como 0. Parâmetro Dados ausentes não disponível se Desenhar estiver configurado como "Pontos" ou "Barra".
Eixo Y	0	ds.axisy.0	0 - (padrão) Esquerda; 1 - Direita.
Deslocamento de tempo	1	ds.timeshift.0	String de tempo válida (e.g. 3600, 1h, etc.). Você pode usar time suffixes . Valores negativos também são permitidos. Padrão: "" (vazio).
Função de agregação	0	ds.aggregate_function.0	0 - (padrão) nenhuma; 1 - minuto; 2 - máximo; 3 - média; 4 - contagem; 5 - somar; 6 - primeiro; 7 - último.
Intervalo da agregação	1	ds.aggregate_interval.0	String de tempo válida (por exemplo 3600, 1h, etc.). You may use time suffixes . Padrão: 1h.
Agregado	0	ds.aggregate_grouping.0	0 - (padrão) Cada item; 1 - Conjunto de dados. Parâmetro Agregado não válido se Função da agregação estiver configurada como "none".

Opções de exibição

Os seguintes parâmetros são suportados para configurar Opções de exibição.

Parâmetro	type	nome	valor
Seleção de dados de histórico	0	source	0 - (padrão) Automático; 1 - Histórico; 2 - Trends.

Período de tempo

Os seguintes parâmetros são suportados para configurar Período de tempo.

Parâmetro	type	nome	valor
Definir período de tempo personalizado	0	graph_time	0 - (padrão) Desativado; 1 - Ativado.
De	1	time_from	String de tempo válida no formato YYYY-MM-DD hh:mm:ss. Relative time period valores de (now, now/d, now/w-1w, etc.) também são suportados.
Para	1	time_to	Padrão: now-1h. String de tempo válida no formato YYYY-MM-DD hh:mm:ss. Relative time period Valores de (now, now/d, now/w-1w, etc.) também são suportados. Padrão: now.

Axes

The following parameters are supported for configuring Axes.

Parameter	type	name	value
Left Y	0	lefty	0 - Disabled; 1 - (default) Enabled.
Right Y	0	righty	Parameter available if Y-axis (in Data set configuration) is set to "Left". 0 - (default) Disabled; 1 - Enabled.
Min	1	lefty_min	Parameter available if Y-axis (in Data set configuration) is set to "Right". Any numeric value. Default: "" (empty).
Max	1	righty_min lefty_max	Any numeric value. Default: "" (empty).
Units (type)	0	righty_max lefty_units	0 - (default) Auto; 1 - Static.
Units (value)	1	righty_units lefty_static_units	Any string value. Default: "" (empty).
X-Axis	0	righty_static_units xaxis	0 - Disabled; 1 - (default) Enabled.

Legend

The following parameters are supported for configuring Legend.

Parameter	type	name	value
Show legend	0	legend	0 - Disabled; 1 - (default) Enabled.
Number of rows	0	legend_lines	Valid values range from 1-5. Default: 1.

Problems

The following parameters are supported for configuring Problems.

Parameter	type	name	value
Show problems	0	show_problems	0 - (default) Disabled; 1 - Enabled.
Selected items only	0	graph_item_problems	0 - Disabled; 1 - (default) Enabled.
Problem hosts	1	problemhosts.0	Host name. Note: The number in the property name references the configured host. To configure multiple hosts, create a dashboard widget field object for each host.
Severity	0	severities	0 - Not classified; 1 - Information; 2 - Warning; 3 - Average; 4 - High; 5 - Disaster. Default: 1, 2, 3, 4, 5 (all enabled).
Problem Tags (the number in the property name (e.g. tags.tag.0) references tag order in the tag evaluation list)	1	problem_name	Note: To configure multiple values, create a dashboard widget field object for each value. Problem event name (case insensitive, full name or part of it).

Parameter	type	name	value
Evaluation type	0	evaltype	0 - (default) And/Or; 2 - Or.
Tag name	1	tags.tag.0	Any string value.
Operator	0	tags.operator.0	Parameter Tag name required if configuring Tags. 0 - Contains; 1 - Equals; 2 - Does not contain; 3 - Does not equal; 4 - Exists; 5 - Does not exist.
Tag value	1	tags.value.0	Parameter Operator required if configuring Tags. Any string value.
			Parameter Tag value required if configuring Tags.

Overrides

The following parameters are supported for configuring Overrides.

Note:

The first number in the property name (e.g. or.hosts.0.0, or.items.0.0) represents the particular data set, while the second number, if present, represents the configured host or item.

Parameter	type	name	value
Host pattern (required)	1	or.hosts.0.0	Host name or pattern (e.g. Zabbix*).
Item pattern (required)	1	or.items.0.0	Item name or pattern (e.g. *: Number of processed *values per second).
Base color	1	or.color.0	Hexadecimal color code (e.g. FF0000).
Width	0	or.width.0	Valid values range from 1-10.
Draw	0	or.type.0	0 - Line; 1 - Points; 2 - Staircase; 3 - Bar.
Transparency	0	or.transparency.0	Valid values range from 1-10.
Fill	0	or.fill.0	Valid values range from 1-10.
Point size	0	or.pointsize.0	Valid values range from 1-10.
Missing data	0	or.missingdatafunc.0	0 - None; 1 - Connected; 2 - Treat as 0.
Y-axis	0	or.axisy.0	0 - Left; 1 - Right.
Time shift	1	or.timeshift.0	Valid time string (e.g. 3600, 1h, etc.). You may use time suffixes . Negative values are allowed.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Graph widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configurando um widget Graph

Configure um widget Gráfico da seguinte forma:

- 4 conjuntos de dados para um total de 9 itens em 1 host.
- Cada conjunto de dados consiste em uma linha que possui uma cor personalizada, largura, transparência e preenchimento.
- O conjunto de dados 4 tem uma agregação configurada.
- Os dados no gráfico são exibidos para um período de tempo das últimas 3 horas.
- Problemas no gráfico são exibidos apenas para os itens configurados.

- O gráfico possui dois eixos Y, dos quais o eixo Y direito exibe valores apenas para o conjunto de dados 4..
- A legenda do gráfico exibe os itens configurados em 2 linhas.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "svggraph",
            "name": "Graph",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 1,
                "name": "ds.hosts.0.0",
                "value": "Zabbix server"
              },
              {
                "type": 1,
                "name": "ds.items.0.0",
                "value": "Zabbix server: Utilization of poller data collector processes, i"
              },
              {
                "type": 1,
                "name": "ds.color.0",
                "value": "FF0000"
              },
              {
                "type": 0,
                "name": "ds.width.0",
                "value": 3
              },
              {
                "type": 0,
                "name": "ds.transparency.0",
                "value": 3
              },
              {
                "type": 0,
                "name": "ds.fill.0",
                "value": 1
              },
              {
                "type": 1,
                "name": "ds.hosts.1.0",
                "value": "Zabbix server"
              },
              {
                "type": 1,
                "name": "ds.items.1.0",
                "value": "Zabbix server: Utilization of trapper data collector processes, i"
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```

{
  "type": 1,
  "name": "ds.color.1",
  "value": "BF00FF"
},
{
  "type": 0,
  "name": "ds.width.1",
  "value": 3
},
{
  "type": 0,
  "name": "ds.transparency.1",
  "value": 3
},
{
  "type": 0,
  "name": "ds.fill.1",
  "value": 1
},
{
  "type": 1,
  "name": "ds.hosts.2.0",
  "value": "Zabbix server"
},
{
  "type": 1,
  "name": "ds.items.2.0",
  "value": "Zabbix server: Utilization of history syncer internal processes,"
},
{
  "type": 1,
  "name": "ds.color.2",
  "value": "0040FF"
},
{
  "type": 0,
  "name": "ds.width.2",
  "value": 3
},
{
  "type": 0,
  "name": "ds.transparency.2",
  "value": 3
},
{
  "type": 0,
  "name": "ds.fill.2",
  "value": 1
},
{
  "type": 1,
  "name": "ds.hosts.3.0",
  "value": "Zabbix server"
},
{
  "type": 1,
  "name": "ds.items.3.0",
  "value": "*: Number of processed *values per second"
},
{
  "type": 1,

```

```

        "name": "ds.color.3",
        "value": "000000"
    },
    {
        "type": 0,
        "name": "ds.transparency.3",
        "value": 0
    },
    {
        "type": 0,
        "name": "ds.fill.3",
        "value": 0
    },
    {
        "type": 0,
        "name": "ds.axisy.3",
        "value": 1
    },
    {
        "type": 0,
        "name": "ds.aggregate_function.3",
        "value": 3
    },
    {
        "type": 1,
        "name": "ds.aggregate_interval.3",
        "value": "1m"
    },
    {
        "type": 0,
        "name": "ds.aggregate_grouping.3",
        "value": 1
    },
    {
        "type": 0,
        "name": "graph_time",
        "value": 1
    },
    {
        "type": 1,
        "name": "time_from",
        "value": "now-3h"
    },
    {
        "type": 0,
        "name": "legend_lines",
        "value": 2
    },
    {
        "type": 0,
        "name": "show_problems",
        "value": 1
    }
}
]
}
],
"userGroups": [
    {
        "usrgrpid": 7,
        "permission": 2
    }
]

```

```

    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

9 Gráfico (clássico)

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget de dashboard permitem configurar o widget **Graph (classic)** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do widget `fields` não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Gráfico (clássico), consulte os requisitos de parâmetro descritos nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Gráfico (clássico).

Parâmetro	type	nome	valor
Intervalo de atualização	0	<code>rf_rate</code>	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Fonte	0	<code>source_type</code>	0 - (padrão) Gráfico; 1 - Gráfico simples.
Gráfico	6	<code>graphid</code>	Graph ID.

Parâmetro Gráfico necessário se Fonte estiver definida como "Gráfico".

Parâmetro	type	nome	valor
Item	4	itemid	Item ID.
Mostrar legenda	0	show_legend	Parâmetro Item necessário se Fonte estiver definido como "Gráfico simples". 0 - Desativado; 1 - (padrão) Ativado.
Item dinâmico	0	dynamic	0 - (padrão) Desativado; 1 - Ativado.

Exemplos

Os seguintes exemplos visam apenas descrever a configuração dos objetos de campo de widget de dashboard para o widget Gráfico (clássico). Para mais informações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget Gráfico (clássico)

Configure um widget Gráfico (clássico) que exiba um gráfico simples para o item "42269".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "graph",
            "name": "Graph (classic)",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "source_type",
                "value": 1
              },
              {
                "type": 4,
                "name": "itemid",
                "value": 42269
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
        "usrgrpId": 7,
        "permission": 2
      }
    ],
    "users": [
      {
        "userid": 1,
```

```

        "permission": 3
    }
],
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

10 Protótipo de gráfico

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo do widget de protótipo de gráfico permitem configurar o widget **Graph prototype** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos `fields` do widget não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Graph prototype, consulte os requisitos de parâmetro delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Protótipo de gráfico.

Parâmetro	type	nome	valor
Intervalo de atualização	0	<code>rf_rate</code>	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Fonte	0	<code>source_type</code>	2 - (padrão) Protótipo de Gráfico; 3 - Protótipo de gráfico simples.
Protótipo de gráfico	7	<code>graphid</code>	Graph prototype ID.
Protótipo item	5	<code>itemid</code>	Parâmetro Protótipo de gráfico necessário se Fonte estiver configurado para "Protótipo de gráfico". Item prototype ID.
Mostrar legenda	0	<code>show_legend</code>	Parâmetro Protótipo de item necessário se Fonte estiver configurado para "Protótipo de gráfico simples". 0 - Desativado; 1 - (padrão) Ativado.

Parâmetro	type	nome	valor
Item dinâmico	0	dynamic	0 - (padrão) Desativado; 1 - Ativado.
Colunas	0	columns	Valores válidos variam de 1-24.
Linhas	0	rows	Padrão: 2. Valores válidos variam de 1-16.
			Padrão: 1.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de protótipo de gráfico para o widget Protótipo de gráfico. Para mais informações sobre como configurar um dashboard, consulte o manual [dashboard.create](#).

Configurando um widget Protótipo de gráfico

Configure um widget Graph prototype que exibe uma grade de 3 gráficos (3 colunas, 1 linha) criados a partir de um protótipo de item (ID: "42316") por descoberta de baixo nível.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "graphprototype",
            "name": "Graph prototype",
            "x": 0,
            "y": 0,
            "width": 16,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "source_type",
                "value": 3
              },
              {
                "type": 5,
                "name": "itemid",
                "value": 42316
              },
              {
                "type": 0,
                "name": "columns",
                "value": 3
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
```

```

        "usrgrpid": 7,
        "permission": 2
    },
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

11 Host availability

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the **Host availability** widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the Host availability widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - (default) 15 minutes.
Host groups	2	groupids	Host group ID.

Note: To configure multiple host groups, create a dashboard widget field object for each host group.

Parameter	type	name	value
Interface type	0	interface_type	0 - None; 1 - Zabbix agent; 2 - SNMP; 3 - IPMI; 4 - JMX. Default: 1, 2, 3, 4 (all enabled). Note: To configure multiple values, create a dashboard widget field object for each value.
Layout	0	layout	0 - (default) Horizontal; 1 - Vertical.
Show hosts in maintenance	0	maintenance	0 - (default) Disabled; 1 - Enabled.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the Host availability widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring a Host availability widget

Configure a Host availability widget that displays availability information (in a vertical layout) for hosts in host group "4" with "Zabbix agent" and "SNMP" interfaces configured.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "hostavail",
            "name": "Host availability",
            "x": 0,
            "y": 0,
            "width": 6,
            "height": 3,
            "view_mode": 0,
            "fields": [
              {
                "type": 2,
                "name": "groupids",
                "value": 4
              },
              {
                "type": 0,
                "name": "interface_type",
                "value": 1
              },
              {
                "type": 0,
                "name": "interface_type",
                "value": 2
              },
              {
                "type": 0,
```

```

        "name": "layout",
        "value": 1
    }
    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

See also

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

12 Valor do item

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo de widget de painel permitem configurar o widget **Item value** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do campo **Widget fields** não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários configurem widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets de forma incorreta. Para garantir a criação ou atualização bem-sucedida do widget Valor do item, consulte os requisitos dos parâmetros descritos nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Valor do Item.

Parâmetro	type	nome	valor
Intervalo de Atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Item (obrigatório)	4	itemid	Item ID.
Show	0	Show	1 - Descrição; 2 - Valor; 3 - Tempo; 4 - Indicador de mudança. Padrão: 1, 2, 3, 4 (todos habilitados). Note: Para configurar múltiplos valores, crie um objeto de campo de widget de painel para cada valor.
Item Dinâmico	0	dynamic	0 - (padrão) Desativado; 1 - Habilitado.
Configuração avançada	0	adv_conf	0 - (padrão) Desativado; 1 - Habilitado.

Configuração avançada

Os seguintes parâmetros são suportados se a Configuração avançada estiver definida como "Habilitada".

Parâmetro	type	nome	valor
Cor de fundo	1	bg_color	Código de cor hexadecimal (por exemplo, FF0000). Padrão: "" (vazio).

Descrição

Os seguintes parâmetros são suportados se a Configuração Avançada estiver definida como "Habilitada", e Mostrar estiver definido como "Description".

Parâmetro	type	nome	valor
Description	1	description	Qualquer valor de string, incluindo macros. Macros suportados: {HOST.*}, {ITEM.*}, {INVENTORY.*}, Macros de usuário. Padrão: {ITEM.NAME}.
Posição Horizontal	0	desc_h_pos	0 - Esquerda; 1 - (padrão) Center; 2 - Direita.
*Posição Vertical	0	desc_v_pos	Dois ou mais elementos (Description, Valor, Tempo) não podem compartilhar a mesma Posição Horizontal e Posição Vertical. 0 - Superior; 1 - Meio; 2 - (Padrão) Inferior.
Tamanho	0	desc_size	Dois ou mais elementos (Description, Valor, Tempo) não podem compartilhar a mesma Posição Horizontal e Posição Vertical. Valores válidos variam de 1-100. Padrão: 15.

Parâmetro	type	nome	valor
Negrito	0	desc_bold	0 - (padrão) Desativado; 1 - Habilitado.
Cor	1	desc_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).

Valor

Os seguintes parâmetros são suportados se a Configuração Avançada estiver definida como "Habilitada", e Mostrar estiver definido como "Valor".

Parâmetro	type	nome	valor
Casas			
Dec- i- mais			
Casas Decimais	0	decimal_places	Valores válidos variam de 1-10. Padrão: 2.
Tamanho	0	decimal_size	Valores válidos variam de 1-100. Padrão: 35.
Posição			
Posição Horizontal	0	value_h_pos	0 - Esquerda; 1 - (padrão) Centro; 2 - Direita.
Posição Vertical	0	value_v_pos	Dois ou mais elementos (Descrição, Valor, Tempo) não podem compartilhar a mesma Posição horizontal e Posição vertical. 0 - Superior; 1 - (padrão) Meio; 2 - Inferior.
Tamanho	0	value_size	Dois ou mais elementos (Descrição, Valor, Tempo) não podem compartilhar a mesma Posição horizontal e Posição vertical. Valores válidos variam de 1-100. Padrão: 45.
Negrito	0	value_bold	0 - Desativado; 1 - (padrão) Habilitado.
Cor	1	value_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).
Unidades			
Unidades (checkbox)	0	units_show	0 - Desativado; 1 - (padrão) Habilitado.
Unidades (valor)	1	units	Qualquer valor de string.
Posição	0	units_pos	0 - Antes do valor; 1 - Acima do valor; 2 - (padrão) Após o valor; 3 - Abaixo do valor.
Tamanho	0	units_size	Valores válidos variam de 1-100. Padrão: 35.
Negrito	0	units_bold	0 - Desativado; 1 - (padrão) Habilitado.
Cor	1	units_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).

Tempo

Os seguintes parâmetros são suportados se a Configuração avançada estiver definida como "Ativada" e Mostrar estiver definido como "Tempo".

Parâmetro	type	nome	valor
Posição Horizontal	0	time_h_pos	0 - Esquerda; 1 - (padrão) Centro; 2 - Direita.
Posição Vertical	0	time_v_pos	Dois ou mais elementos (Descrição, Valor, Tempo) não podem compartilhar a mesma Posição horizontal e Posição vertical. 0 - (padrão) Dois ou mais elementos (Descrição, Valor, Tempo) não podem compartilhar a mesma Posição horizontal e Posição vertical
Tamanho	0	time_size	Valores válidos variam de 1-100.
Negrito	0	time_bold	Padrão: 15. 0 - (padrão) Desativado; 1 - Habilitado.
Cor	1	time_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).

Indicador de mudança

Os seguintes parâmetros são suportados se a Configuração avançada estiver definida como "Ativada" e Mostrar estiver definido como "Indicador de Mudança".

Parâmetro	type	nome	valor
Cor da seta de aumento ↑	1	up_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).
Cor da seta de redução ↓	1	down_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (vazio).
Change indicator ↕ color	1	updown_color	Código de cor hexadecimal (Por exemplo, FF0000). Padrão: "" (empty).

Exemplos

Os seguintes exemplos têm como objetivo descrever apenas a configuração dos objetos de campo do widget de Valor do item para o painel de controle. Para mais informações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget de Valor do item

Configure um widget de Valor do item que exiba o valor do item para o item "42266" (Disponibilidade do agente Zabbix). Além disso, ajuste visualmente o widget com várias opções avançadas.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "item",
            "name": "Item value",
            "x": 0,
```

```

"y": 0,
"width": 4,
"height": 3,
"view_mode": 0,
"fields": [
  {
    "type": 4,
    "name": "itemid",
    "value": 42266
  },
  {
    "type": 0,
    "name": "show",
    "value": 1
  },
  {
    "type": 0,
    "name": "show",
    "value": 2
  },
  {
    "type": 0,
    "name": "show",
    "value": 3
  },
  {
    "type": 0,
    "name": "adv_conf",
    "value": 1
  },
  {
    "type": 1,
    "name": "bg_color",
    "value": "D1C4E9"
  },
  {
    "type": 1,
    "name": "description",
    "value": "Agent status"
  },
  {
    "type": 0,
    "name": "desc_h_pos",
    "value": 0
  },
  {
    "type": 0,
    "name": "desc_v_pos",
    "value": 0
  },
  {
    "type": 0,
    "name": "desc_bold",
    "value": 1
  },
  {
    "type": 1,
    "name": "desc_color",
    "value": "F06291"
  },
  {
    "type": 0,

```

```

        "name": "value_h_pos",
        "value": 0
    },
    {
        "type": 0,
        "name": "value_size",
        "value": 25
    },
    {
        "type": 1,
        "name": "value_color",
        "value": "FFFF00"
    },
    {
        "type": 0,
        "name": "units_show",
        "value": 0
    },
    {
        "type": 0,
        "name": "time_h_pos",
        "value": 2
    },
    {
        "type": 0,
        "name": "time_v_pos",
        "value": 2
    },
    {
        "type": 0,
        "name": "time_size",
        "value": 10
    },
    {
        "type": 0,
        "name": "time_bold",
        "value": 1
    },
    {
        "type": 1,
        "name": "time_color",
        "value": "9FA8DA"
    }
}

    ]
}

    ],
    "userGroups": [
        {
            "usrgrpId": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userId": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",

```

```
"id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

13 Mapa

Descrição

Estes parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo do widget de dashboard permitem configurar o **Map** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do `widgetfields` não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os Widgets integrados **built-in widgets** e criem Widgets personalizados **custom widgets**, mas também introduz o risco de criar e atualizar Widgets de forma incorreta. Para garantir a criação ou atualização bem-sucedida do widget Mapa, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o Widget Map.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - (padrão) 15 minutos.
Tipo de fonte	0	source_type	1 - (padrão) Mapa; 2 - Árvore de navegação do mapa.
Mapa	8	sysmapid	ID do Map .
Referência do widget vinculado	1	filter_widget_reference	Parâmetro Map necessário se Tipo de fonte estiver configurado como "Mapa". Valor válido do parâmetro Referência do widget Map navigation tree . Parâmetro Referência do widget vinculado necessário se Tipo de fonte estiver configurado como "Árvore de navegação do mapa".

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de dashboard para o widget Mapa. Para obter mais informações sobre a configuração de um dashboard, consulte [dashboard.create](#).

Configurando um widget Map

Configure um widget Map que exibe o mapa "1".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "map",
            "name": "Map",
            "x": 0,
            "y": 0,
            "width": 18,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 8,
                "name": "sysmapid",
                "value": 1
              }
            ]
          }
        ]
      }
    ],
    "userGroups": [
      {
        "usrgrpid": 7,
        "permission": 2
      }
    ],
    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

Configurando um widget Mapa vinculado

Configure um widget Mapa vinculado a um widget **Map navigation tree**.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "map",
            "name": "Map",
            "x": 0,
            "y": 5,
            "width": 18,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "source_type",
                "value": 2
              },
              {
                "type": 1,
                "name": "filter_widget_reference",
                "value": "ABCDE"
              }
            ]
          }
        ],
        {
          "type": "navtree",
          "name": "Map navigation tree",
          "x": 0,
          "y": 0,
          "width": 6,
          "height": 5,
          "view_mode": 0,
          "fields": [
            {
              "type": 1,
              "name": "navtree.name.1",
              "value": "Element A"
            },
            {
              "type": 1,
              "name": "navtree.name.2",
              "value": "Element B"
            },
            {
              "type": 1,
              "name": "navtree.name.3",
              "value": "Element C"
            },
            {
              "type": 1,
              "name": "navtree.name.4",
              "value": "Element A1"
            }
          ]
        }
      ]
    }
  }
}
```

```

        "type": 1,
        "name": "navtree.name.5",
        "value": "Element A2"
    },
    {
        "type": 1,
        "name": "navtree.name.6",
        "value": "Element B1"
    },
    {
        "type": 1,
        "name": "navtree.name.7",
        "value": "Element B2"
    },
    {
        "type": 0,
        "name": "navtree.parent.4",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.parent.5",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.parent.6",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.parent.7",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.1",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.order.2",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.3",
        "value": 3
    },
    {
        "type": 0,
        "name": "navtree.order.4",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.order.5",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.6",

```

```

        "value": 1
      },
      {
        "type": 0,
        "name": "navtree.order.7",
        "value": 2
      },
      {
        "type": 8,
        "name": "navtree.sysmapid.6",
        "value": 1
      },
      {
        "type": 1,
        "name": "reference",
        "value": "ABCDE"
      }
    ]
  },
  "userGroups": [
    {
      "usrgrpid": 7,
      "permission": 2
    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)
- [Map navigation tree](#)

14 Árvore de navegação do mapa

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo do widget do tipo Map navigation tree

permitem configurar o widget [Map navigation tree] (/manual/web_interface/frontend_sections/monitoring/dashboard/widgets/map_tree) nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos `fields` do widget não são validadas durante a criação ou atualização de um painel. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Map navigation tree, consulte os requisitos de parâmetro delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Map navigation tree.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - (padrão) 15 minutos.
Mostrar mapas indisponíveis	1	show_unavailable	0 - (padrão) Desativado; 1 - Ativado.
Referência	1	reference	Qualquer valor de string consistindo de 5 caracteres (por exemplo, ABCDE, JBPNL, etc.). O valor do parâmetro da Referência é usado no widget Map (Referência de widget vinculado) para vincular com o widget Map navigation tree.

Os seguintes parâmetros são suportados para configurar os elementos da árvore de navegação de mapas.

Parâmetro	type	nome	valor
Nome	1	navtree.name.1	Qualquer valor de string.
Mapa vinculado	8	navtree.sysmapid.1	Note: O número no nome da propriedade define o número do elemento. Map ID.
Parâmetros para criar hierarquia de elementos	0	navtree.parent.1	Note: O número do nome da propriedade referencia o elemento ao qual o mapa está vinculado. Número do elemento pai.
	0	navtree.order.1	Note: O número do nome da propriedade referencia o elemento filho. O valor da propriedade referencia o elemento pai. Posição do elemento na árvore de navegação do mapa.
			Note: O número no nome da propriedade referencia o número do elemento. O valor da propriedade referencia a posição do elemento na árvore de navegação do mapa. A posição do elemento pai é determinada dentro de toda a árvore de navegação do mapa. A posição do elemento filho é determinada dentro do elemento pai.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de painel para o widget Map navigation tree. Para mais configurações sobre como configurar o dashboard, consulte o manual **dashboard.create**.

Configurando um widget Map navigation tree

Configure um widget Map navigation tree que exibe a seguinte árvore de navegação de mapas:

- Elemento A
 - Elemento A1

- Elemento A2
- Elemento B
 - Elemento B1 (contém mapa vinculado "1" que pode ser exibido em um **linked Map widget**)
 - Elemento B2
- Elemento C

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "navtree",
            "name": "Map navigation tree",
            "x": 0,
            "y": 0,
            "width": 6,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 1,
                "name": "navtree.name.1",
                "value": "Element A"
              },
              {
                "type": 1,
                "name": "navtree.name.2",
                "value": "Element B"
              },
              {
                "type": 1,
                "name": "navtree.name.3",
                "value": "Element C"
              },
              {
                "type": 1,
                "name": "navtree.name.4",
                "value": "Element A1"
              },
              {
                "type": 1,
                "name": "navtree.name.5",
                "value": "Element A2"
              },
              {
                "type": 1,
                "name": "navtree.name.6",
                "value": "Element B1"
              },
              {
                "type": 1,
                "name": "navtree.name.7",
                "value": "Element B2"
              },
              {
                "type": 0,

```

```

        "name": "navtree.parent.4",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.parent.5",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.parent.6",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.parent.7",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.1",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.order.2",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.3",
        "value": 3
    },
    {
        "type": 0,
        "name": "navtree.order.4",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.order.5",
        "value": 2
    },
    {
        "type": 0,
        "name": "navtree.order.6",
        "value": 1
    },
    {
        "type": 0,
        "name": "navtree.order.7",
        "value": 2
    },
    {
        "type": 8,
        "name": "navtree.sysmapid.6",
        "value": 1
    },
    {
        "type": 1,
        "name": "reference",
        "value": "ABCDE"
    }

```

```

    }
  ]
},
"userGroups": [
  {
    "usrgrpId": 7,
    "permission": 2
  }
],
"users": [
  {
    "userId": 1,
    "permission": 3
  }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)
- [Map](#)

15 Texto simples

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo do widget de dashboard permitem configurar o widget **Plain text** nos métodos `dashboard.create` e `dashboard.update` /

Attention:

As propriedades do widget `fields` não são validadas durante a criação e atualização de um dashboard. Isso permite que os usuários modifiquem os Widgets integrados **built-in widgets** e criem Widgets personalizados **custom widgets**, mas também introduz o risco de criar e atualizar Widgets de forma incorreta. Para garantir a criação ou atualização bem-sucedida do widget Texto simples, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Texto simples.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Itens (obrigatório)	4	itemids	ID do Item . Observe: Para configurar vários itens, crie um objeto de campo de widget de dashboard para cada item.
Localização dos itens	0	style	0 - (padrão) Esquerda; 1 - Superior.
Mostrar linhas	0	show_lines	Valores válidos variam de 1-100.
Mostrar texto como HTML	0	show_as_html	Padrão: 25. 0 - (padrão) Desativado; 1 - Habilitado.
Item dinâmico	0	dynamic	0 - (padrão) Desativado; 1 - habilitado.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de dashboard para o widget Texto simples. Para obter mais informações sobre a configuração de um dashboard, consulte [dashboard.create](#).

Configurando um widget Texto simples

Configure um widget Texto Simples que exibe os dados mais recentes para os itens "42269" e "42253". Além disso, configure os nomes dos itens para serem localizados no topo das colunas de dados e exiba apenas 15 linhas de dados.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "plaintext",
            "name": "Plain text",
            "x": 0,
            "y": 0,
            "width": 6,
            "height": 3,
            "view_mode": 0,
            "fields": [
              {
                "type": 4,
                "name": "itemids",
                "value": 42269
              },
              {
                "type": 4,
                "name": "itemids",
                "value": 42253
              }
            ]
          }
        ]
      }
    ]
  }
}
```

```

        "type": 0,
        "name": "style",
        "value": 1
    },
    {
        "type": 0,
        "name": "show_lines",
        "value": 15
    }
]
}
],
"userGroups": [
    {
        "usrgrpid": 7,
        "permission": 2
    }
],
"users": [
    {
        "userid": 1,
        "permission": 3
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

16 Hosts com problemas

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo de widget dashboard permitem configurar o widget **Problem hosts** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do widget `fields` não são válidas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Hosts com problemas, consulte os requisitos de parâmetro descritos nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget Hosts com problemas.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem intervalo; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de host	2	groupids	Host group ID. Note: Para configurar vários grupos de host, crie um objeto de campo de widget de dashboard para cada grupo de host.
Excluir grupos de host	2	exclude_groupids	Host group ID. Note: Para excluir vários grupos de host, crie um objeto de campo de widget de dashboard para cada grupo de host.
Hosts	3	hostids	Host ID. Note: Para configurar vários hosts, crie um objeto de campo de widget de dashboard para cada host. Para vários hosts, o parâmetro Grupos de host deve ser configurado com pelo menos um grupo de host ao qual os hosts configurados pertencem.
Problema	1	problem	Problema event name (sensível a maiúsculas e minúsculas, nome completo ou parte dele).
Gravidade	0	severities	0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Médio; 4 - Alto; 5 - Desastre. Padrão 1, 2, 3, 4, 5 (todos habilitados). Note: Para configurar vários valores, crie um objeto de campo de widget de dashboard para cada valor.

Parâmetro	type	nome	valor
Tags (o número no nome da pro- priedade (por ex- em- plo, tags.tag.0) az refer- ên- cia à or- dem da tag na lista de avali- ação de tags)			
Tipo avaliação	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Tag name	1	tags.tag.0	Qualquer valor de string.
Operador	0	tags.operator.0	Parâmetro Nome da tag necessário se estiver configurando Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não contém; 4 - Existe; 5 - Não existe.
Valor da tag	1	tags.value.0	Parâmetro Operador necessário se estiver configurando Tags. Qualquer valor de string.
Mostrar prob- le- mas suprim- i- dos	0	show_suppressed	Parâmetro Valor da tag necessário se estiver configurando Tags. 0 - (padrão) Desativado; 1 - Ativado.
Ocultar gru- pos sem prob- le- mas	0	hide_empty_groups	0 - (padrão) Desativado; 1 - Ativado.

Parâmetro	type	nome	valor
Exibição de problema	0	ext_ack	0 - (padrão) All; 1 - Apenas não reconhecidos; 2 - Separados.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo de widget de dashboard para o widget Hosts com problemas. Para mais informações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget Hosts com problemas

Configure um widget Hosts com problemas que exibe hosts dos grupos de hosts "2" e "4" que possuem problemas com um nome que inclui a string "CPU" e que possuem as seguintes gravidades: "Aviso", "Médio", "Alto", "Desastre"

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "problemhosts",
            "name": "Problem hosts",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 2,
                "name": "groupids",
                "value": 2
              },
              {
                "type": 2,
                "name": "groupids",
                "value": 4
              },
              {
                "type": 1,
                "name": "problem",
                "value": "cpu"
              },
              {
                "type": 0,
                "name": "severities",
                "value": 2
              },
              {
                "type": 0,
                "name": "severities",
                "value": 3
              },
              {
                "type": 0,
```

```

        "name": "severities",
        "value": 4
    },
    {
        "type": 0,
        "name": "severities",
        "value": 5
    }
]
}
],
"userGroups": [
    {
        "usrgrpId": 7,
        "permission": 2
    }
],
"users": [
    {
        "userId": 1,
        "permission": 3
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

17 Problemas

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo do widget de dashboard permitem configurar o widget **Problemas** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades do widget `fields` não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os Widgets integrados **built-in widgets** e criem Widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets de forma incorreta. Para garantir a criação ou atualização bem-sucedida do widget **Problemas**, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para widget Problemas

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Mostrar	0	show	1 - (padrão) Problemas recentes; 2 - Histórico; 3 - Problemas.
Grupos de host	2	groupids	ID do Host group . Observe: crie um objeto de campo de widget de painel para cada grupo de host.
Excluir grupos de host	2	exclude_groupids	ID do Host group . Observe: Para excluir vários grupos de hosts, crie um objeto de campo de widget de painel para cada grupo de hosts.
Hosts	3	hostids	ID do Host . Observe: Para configurar vários hosts, crie um objeto de campo de widget de painel para cada host. Para múltiplos hosts, o parâmetro Grupos de hosts deve estar não configurado ou configurado com pelo menos um grupo de host ao qual os hosts configurados pertencem.
Problema	1	problem	Nome do evento problema event name (maiúsculas e minúsculas, nome completo ou parte dele).
Severidade	0	severities	0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Médio; 4 - Alto; 5 - Desastre. Padrão: 1, 2, 3, 4, 5 (todos habilitados). Observe: Para configurar múltiplos valores, crie um objeto de campo de widget de painel para cada valor.

Parâmetro	type	nome	valor
Tags (o número no nome da pro- priedade (por ex- em- plo, tags.tag.0) refere- se à or- dem da tag na lista de avali- ação de tags)			
Evaluation type	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Nome da tag	1	tags.tag.0	Qualquer valor de string.
Operador	0	tags.operator.0	Parâmetro Nome da tag necessário se estiver configurando Tags. 0 - Contains; 1 - Igual; 2 - Não é igual; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Valor da tag	1	tags.value.0	Parâmetro Operador necessário se estiver configurando Tags. Qualquer valor de string.
Mostrar tags	0	show_tags	Parâmetro Valor de tag necessário se estiver configurando Tags. 0 - (padrão) Nenhuma; 1 - 1; 2 - 2; 3 - 3.
Nome da tag (for- mato)	0	tag_name_format	0 - (padrão) Inteiro; 1 - Shortened; 2 - Nenhum.
Exibir pro- priedade da tag	1	tag_priority	Parâmetro Nome da tag (formato) não disponível se Mostrar tags estiver configurado como "Nenhum". Lista separada por vírgulas de tags. Parâmetro Tag display priority não disponível se Mostrar tags estiver configurado como "None".

Parâmetro	type	nome	valor
Mostrar op- er- a- tional data	0	show_opdata	0 - (padrão) Nenhum; 1 - Separadamente; 2 - Com o nome do problema.
Mostrar prob- le- mas suprim- i- dos	0	show_suppressed	0 - (padrão) Desabilitado; 1 - Habilitado.
Show so- mente de- scon- hecido	0	unacknowledged	0 - (padrão) Desabilitado; 1 - Habilitado.
Ordenar en- tradas por	0	sort_triggers	1 - Severidade (decrecente); 2 - Host (crescente); 3 - Horário (crescente); 4 - (padrão) Horário (decrecente); 13 - Severidade (crescente); 14 - Host (decrecente); 15 - Problema (crescente); 16 - Problema (decrecente).
Mostrar linha do tempo	0	show_timeline	Para todos valores, exceto "Horário (decrecente)" e "Horário (crescente)", o parâmetro Mostrar timeline deve estar configurado como "Desabilitado". 0 - Desabilitado; 1 - (padrão) Habilitado.
Mostrar lin- has	0	show_lines	Parâmetro Mostrar linha do tempo disponível se Ordenar entradas por estiver configurado para "Horário (decrecente)" ou "Horário (crescente)". Valores válidos variam de 1-100. Padrão: 25.

Exemplos

Os seguintes exemplos têm como objetivo descrever apenas a configuração dos objetos de campo de widget do painel para o widget Problemas. Para obter mais informações sobre como configurar um painel, consulte [dashboard.create](#).

Configurando um widget Problems

Configure um widget Problemas que exibe problemas para o grupo de hosts "4" que satisfaçam as seguintes condições:

- Problemas que têm uma tag com o nome "scope" que contém os valores "performance" ou "availability", ou "capacity".
- Problemas que têm as seguintes severidades: "Aviso", "Média", "Alto", "Desastre".

Além disso, configure o widget para mostrar tags e dados operacionais.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
```

```

"pages": [
  {
    "widgets": [
      {
        "type": "problems",
        "name": "Problems",
        "x": 0,
        "y": 0,
        "width": 12,
        "height": 5,
        "view_mode": 0,
        "fields": [
          {
            "type": 2,
            "name": "groupids",
            "value": 4
          },
          {
            "type": 1,
            "name": "tags.tag.0",
            "value": "scope"
          },
          {
            "type": 0,
            "name": "tags.operator.0",
            "value": 0
          },
          {
            "type": 1,
            "name": "tags.value.0",
            "value": "performance"
          },
          {
            "type": 1,
            "name": "tags.tag.1",
            "value": "scope"
          },
          {
            "type": 0,
            "name": "tags.operator.1",
            "value": 0
          },
          {
            "type": 1,
            "name": "tags.value.1",
            "value": "availability"
          },
          {
            "type": 1,
            "name": "tags.tag.2",
            "value": "scope"
          },
          {
            "type": 0,
            "name": "tags.operator.2",
            "value": 0
          },
          {
            "type": 1,
            "name": "tags.value.2",
            "value": "capacity"
          }
        ]
      }
    ]
  }
]

```

```

        {
            "type": 0,
            "name": "severities",
            "value": 2
        },
        {
            "type": 0,
            "name": "severities",
            "value": 3
        },
        {
            "type": 0,
            "name": "severities",
            "value": 4
        },
        {
            "type": 0,
            "name": "severities",
            "value": 5
        },
        {
            "type": 0,
            "name": "show_tags",
            "value": 1
        },
        {
            "type": 0,
            "name": "show_opdata",
            "value": 1
        }
    ]
}

    ],
    "userGroups": [
        {
            "usrgrpid": 7,
            "permission": 2
        }
    ],
    "users": [
        {
            "userid": 1,
            "permission": 3
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    },
    "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

18 Problemas por severidade

Descrição

Esses parâmetros e os possíveis valores de propriedade para os respectivos objetos de campo de widget do dashboard permitem configurar o widget **Problems by severity** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos do widget `fields` não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem os **built-in widgets** e criem **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Problemas por gravidade, consulte os requisitos de parâmetros descritos nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para widget Problemas por severidade.

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de host	2	groupids	ID do Host group . Note: Para configurar vários grupos de host, crie um objeto de campo de widget de painel para cada grupo de hosts.
*Excluir grupos de host	2	exclude_groupids	ID do Host group . Note: Para excluir vários grupos de host, crie um objeto de campo de widget de painel para cada grupo de hosts.
Hosts	3	hostids	ID do Host . Note: Para configurar vários hosts, Um objeto de campo de widget de painel para cada grupo de hosts. Para vários hosts, o parâmetro Grupos de Hosts deve estar ou não configurado de forma alguma, ou configurado com pelo menos um grupo de hosts ao qual os hosts configurados pertencem.
Problema	1	problem	Nome do evento problema event name (maiúscula minúscula, nome completo ou parte dele).
Severidade	0	severities	0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Média; 4 - Alto; 5 - Desastre. Padrão: vazio (todos habilitados). Note: Para configurar vários valores, crie Um objeto de campo de widget de painel para cada grupo de hosts.

Parâmetro	type	nome	valor
Tags (o número no nome da pro- priedade (por ex- em- plo, tags.tag.0) refere- se à or- dem de clas- si- fi- cação na lista de avali- ação de tags			
Evaluation type	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Tag name	1	tags.tag.0	Qualquer valor de tag.
Operator	0	tags.operator.0	Parâmetro Nome da tag necessário se estiver configurando Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Tag value	1	tags.value.0	Parâmetro Operador necessário se estiver configurando Tags. Qualquer valor de string.
Mostrar	0	show_type	Parâmetro Valor da Tag necessário se estiver configurando Tags. 0 - (padrão) Grupos de host; 1 - Total.
Layout	0	layout	0 - (padrão) Horizontal; 1 - Vertical.
*Mostrar da- dos op- era- cionais	0	show_opdata	Parâmetro Layout não disponível se Mostrar estiver configurado como "Host groups". 0 - (padrão) Nenhum; 1 - Separadamente; 2 - Com nome do problema.

Parâmetro	type	nome	valor
Mostrar prob- le- mas suprim- i- dos	0	show_suppressed	0 - (padrão) Desabilitado; 1 - Habilitado.
Ocultar gru- pos sem prob- le- mas	0	hide_empty_groups	0 - (padrão) Desabilitado; 1 - Habilitado. Parâmetro Ocultar grupos sem problemas não disponível se Mostrar estiver configurado como "Total".
Exibir prob- lema	0	ext_ack	0 - (padrão) Todos; 1 - Somente não reconhecidos; 2 - Separados.
Mostrar time- line	0	show_timeline	0 - Desabilitada; 1 - (padrão) Habilitada.

Exemplos

Os exemplos a seguir têm como objetivo apenas descrever a configuração dos objetos de campo do widget do painel para o widget Problems por gravidade. Para mais configurações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget Problemas por gravidade

Configure um widget Problemas por gravidade que exiba o número total de problemas para todos grupos de host.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "problemsbysv",
            "name": "Problems by severity",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "show_type",
                "value": 1
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
    {
```

Resposta:

Consulte também

- ## 19 Relatório SLA

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo do widget de relatório de SLA permitem configurar o widget de relatório SLA **SLA report** nos métodos `dashboard.create`, `dashboard.update`.

Attention:

As propriedades dos campos do widget `fields` não são validadas durante a criação ou atualização de um dashboard. Isso permite que os usuários modifiquem widgets integrados **built-in widgets** e criem widgets personalizados **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget de Relatório de SLA, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget de Relatório SLA.

1243

Parâmetro	type	nome	valor
Mostrar períodos	0	show_periods	Valores válidos variam de 1-100.
De	1	date_from	Padrão: 20. String de data válida no formato YYYY-MM-DD. Relative dates com modificadores d, w, M, y (por exemplo, now, now/d, now/w-1w, etc.) são suportadas.
Para	1	date_to	String de data válida no formato YYYY-MM-DD. Relative dates com modificadores d, w, M, y (e.g. now, now/d, now/w-1w, etc.) são suportadas.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de Relatório de SLA para o painel. Para mais informações sobre como configurar um dashboard, consulte [dashboard.create](#).

Configurando um widget de Relatório de SLA

Configure um widget de Relatório de SLA que exiba o relatório de SLA para o SLA "4" serviço "2" para o período dos últimos 30 dias.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "slareport",
            "name": "SLA report",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 10,
                "name": "slaid",
                "value": 4
              },
              {
                "type": 9,
                "name": "serviceid",
                "value": 2
              },
              {
                "type": 1,
                "name": "date_from",
                "value": "now-30d"
              },
              {
                "type": 1,
                "name": "date_to",
                "value": "now"
              }
            ]
          }
        ]
      }
    ]
  }
}
```



```

    ]
  },
  "userGroups": [
    {
      "usrgrpid": 7,
      "permission": 2
    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

20 System information

Description

These parameters and the possible property values for the respective dashboard widget field objects allow to configure the **System Information** widget in `dashboard.create` and `dashboard.update` methods.

Parameters

The following parameters are supported for the System Information widget.

Parameter	type	name	value
Refresh interval	0	rf_rate	0 - No refresh; 10 - 10 seconds; 30 - 30 seconds; 60 - 1 minute; 120 - 2 minutes; 600 - 10 minutes; 900 - (default) 15 minutes.
Show	0	info_type	0 - (default) System stats; 1 - High availability nodes.

Examples

The following examples aim to only describe the configuration of the dashboard widget field objects for the System information widget. For more information on configuring a dashboard, see [dashboard.create](#).

Configuring a System information widget

Configure a System information widget that displays system stats with a refresh interval of 10 minutes.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "systeminfo",
            "name": "System information",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 0,
                "name": "rf_rate",
                "value": 600
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
    {
      "usrgrpid": 7,
      "permission": 2
    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

See also

- `Dashboard widget field`
- `dashboard.create`
- `dashboard.update`

21 Top hosts

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget permitem configurar o **Top Hosts** nos métodos `dashboard.create` e `dashboard.update`.

Parâmetros

Os seguintes parâmetros são suportados para o widget Top Hosts.

Parameter	type	nome	valor
Refresh interval	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Host groups	2	groupids	Host group ID. Note: Para configurar vários host groups, crie um objeto de campo de widget de painel para cada host.
Hosts	3	hostids	Host ID. Note: Para configurar múltiplos hosts, crie um objeto de campo de widget de painel para cada host. Para múltiplos hosts, o parâmetro Host groups não deve ser configurado ou configurado com pelo menos um host group ao qual os hosts configurados pertencem.
Host Tags (the number in the property name (e.g. tags.tag.0) references tag order in the tag evaluation list)	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.

Parameter	type	nome	valor
Tag name	1	tags.tag.0	Qualquer valor da string.
Operator	0	tags.operator.0	Parâmetro Nome da Tag é necessário se estiver configurando as Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Tag value	1	tags.value.0	Parâmetro Operador é necessário se estiver configurando as Tags. Qualquer valor da string.
Columns (see below)			Parâmetro Valor da tag é necessário se estiver configurando as Tags.
Order	0	order	2 - (padrão) Top N; 3 - Botão N.
Order column	0	column	Valor numérico da coluna configurada
Host count	0	count	Valores válidos variam entre 1-100. Padrão: 10.

Colunas

As colunas possuem parâmetros comuns e parâmetros adicionais dependendo da configuração do parâmetro Data.

Note:

Para todos os parâmetros relacionados a colunas, o número no nome da propriedade (por exemplo, columnsn.name.0) refere-se a uma coluna para qual o parâmetro está configurado.

Os seguintes parâmetros são suportados por todas as colunas.

Parameter	type	nome	valor
Name	1	columns.name.0	Qualquer valor da string.
Data (obrigatório)	0	columns.data.0	1 - Valor do item; 2 - Nome do host; 3 - Texto.
Base color (obrigatório)	1	columns.base_color.0	Código de cor Hexadecimal (por exemplo, FF0000).

Valor do item

Os seguintes parâmetros são suportados se Data estiver configurado como "Valor do item".

Note:

O primeiro número no nome da propriedade Thresholds (por exemplo, columnsthresholds.color.0.0) refere-se à coluna para qual os thresholds estão configurados, enquanto o segundo número faz referência à posição dos thresholds em uma lista, ordenada em ordem crescente. Contudo, se thresholds estiverem configurados em uma ordem diferente, os valores serão ordenados em ordem crescente após a atualização da configuração do widget no frontend do Zabbix.
(por exemplo, "threshold.threshold.0": "5" → "threshold.threshold.0": "1";
"threshold.threshold.1": "1" → "threshold.threshold.1": "5").

Parameter	type	nome	valor
Item	1	columns.item.0	Valide o nome do item.
Time shift (required)	1	columns.timeshift.0	Valide o valor numérico ou o tempo da string (por exemplo, 3600 ou 1h). Você pode usar time suffixes . Valores negativos são permitidos.
Aggregation func- tion	0	columns.aggregate_function.0	nenhum; 1 - min; 2 - max; 3 - média; 4 - contar; 5 - somar; 6 - primeiro; 7 - último.
Aggregation in- ter- val	1	columns.aggregate_interval.0	Valide o tempo da string (por exemplo, 3600, 1h, etc.). Você pode usar time suffixes . Parâmetro Intervalo de agregação indisponível se Função da agregação estiver configurado como none.
Display	0	columns.display.0	Padrão: 1h. 1 - (padrão) As is; 2 - Barra; 3 - Indicadores.
Min	1	columns.min.0	Qualquer valor numérico. Parâmetro Min indisponível se Display estiver configurado como "As is".
Max	1	columns.max.0	Qualquer valor numérico. Parâmetro Max indisponível se Display estiver configurado como "As is".
History data	0	columns.history.0	1 - (padrão) Automático; 2 - Histórico; 3 - Trends.
Thresholds Color	1	columnsthresholds.color.0	Qualquer código de cor Hexadecimal (por exemplo, FF0000). Padrão: "" (vazio).
Threshold	1	columnsthresholds.threshold.0	Qualquer valor da string.

Texto

Os seguintes parâmetros são suportados se Data estiver configurado como "Texto".

Parameter	type	name	valor
Text	1	columns.text.0	Qualquer valor da string, incluindo macros Macros suportados: {HOST.*}, {INVENTORY.*}. Parâmetro Texto obrigatório se Data estiver configurado como "Texto".

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget do painel para o Top hosts widget. Para mais informações sobre como configurar um painel, consulte [dashboard.create](#).

Configurando um widget Top hosts

Configure um widget Top hosts que exibe os principais hosts por utilização da CPU no grupo de hosts "4" Além disso, configure as colunas personalizadas a seguir: "Nome do host", "Utilização", "1m avg", "5m avg", "15m avg", "Processos".

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "tophosts",
            "name": "Top hosts",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 2,
                "name": "groupids",
                "value": 4
              },
              {
                "type": 1,
                "name": "columns.name.0",
                "value": ""
              },
              {
                "type": 0,
                "name": "columns.data.0",
                "value": 2
              },
              {
                "type": 1,
                "name": "columns.base_color.0",
                "value": "FFFFFF"
              },
              {
                "type": 1,
                "name": "columns.timeshift.0",
                "value": ""
              },
              {
                "type": 1,
                "name": "columns.name.1",
                "value": "Utilization"
              },
              {
                "type": 0,
                "name": "columns.data.1",
                "value": 1
              },
              {
                "type": 1,
                "name": "columns.base_color.1",
                "value": "4CAF50"
              },
              {
                "type": 1,
                "name": "columns.timeshift.1",

```

```

        "value": ""
    },
    {
        "type": 1,
        "name": "columns.item.1",
        "value": "CPU utilization"
    },
    {
        "type": 0,
        "name": "columns.display.1",
        "value": 3
    },
    {
        "type": 1,
        "name": "columns.min.1",
        "value": "0"
    },
    {
        "type": 1,
        "name": "columns.max.1",
        "value": "100"
    },
    {
        "type": 1,
        "name": "columnsthresholds.color.1.0",
        "value": "FFFF00"
    },
    {
        "type": 1,
        "name": "columnsthresholds.threshold.1.0",
        "value": "50"
    },
    {
        "type": 1,
        "name": "columnsthresholds.color.1.1",
        "value": "FF8000"
    },
    {
        "type": 1,
        "name": "columnsthresholds.threshold.1.1",
        "value": "80"
    },
    {
        "type": 1,
        "name": "columnsthresholds.color.1.2",
        "value": "FF4000"
    },
    {
        "type": 1,
        "name": "columnsthresholds.threshold.1.2",
        "value": "90"
    },
    {
        "type": 1,
        "name": "columns.name.2",
        "value": "1m avg"
    },
    {
        "type": 0,
        "name": "columns.data.2",
        "value": 1
    },
    },

```

```

{
  "type": 1,
  "name": "columns.base_color.2",
  "value": "FFFFFF"
},
{
  "type": 1,
  "name": "columns.timeshift.2",
  "value": ""
},
{
  "type": 1,
  "name": "columns.item.2",
  "value": "Load average (1m avg)"
},
{
  "type": 1,
  "name": "columns.name.3",
  "value": "5m avg"
},
{
  "type": 0,
  "name": "columns.data.3",
  "value": 1
},
{
  "type": 1,
  "name": "columns.base_color.3",
  "value": "FFFFFF"
},
{
  "type": 1,
  "name": "columns.timeshift.3",
  "value": ""
},
{
  "type": 1,
  "name": "columns.item.3",
  "value": "Load average (5m avg)"
},
{
  "type": 1,
  "name": "columns.name.4",
  "value": "15m avg"
},
{
  "type": 0,
  "name": "columns.data.4",
  "value": 1
},
{
  "type": 1,
  "name": "columns.base_color.4",
  "value": "FFFFFF"
},
{
  "type": 1,
  "name": "columns.timeshift.4",
  "value": ""
},
{
  "type": 1,

```



```

        "name": "columns.item.4",
        "value": "Load average (15m avg)"
    },
    {
        "type": 1,
        "name": "columns.name.5",
        "value": "Processes"
    },
    {
        "type": 0,
        "name": "columns.data.5",
        "value": 1
    },
    {
        "type": 1,
        "name": "columns.base_color.5",
        "value": "FFFFFF"
    },
    {
        "type": 1,
        "name": "columns.timeshift.5",
        "value": ""
    },
    {
        "type": 1,
        "name": "columns.item.5",
        "value": "Number of processes"
    },
    {
        "type": 0,
        "name": "column",
        "value": 1
    }
    ]
}
],
"userGroups": [
    {
        "usrgrpid": 7,
        "permission": 2
    }
],
"users": [
    {
        "userid": 1,
        "permission": 3
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "3"
        ]
    }
}

```

```

    },
    "id": 1
  }

```

Veja também

-Campo widget do dashboard -`dashboard.create` -`dashboard.update`

22 Visão geral do trigger

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget de painel respectivos permitem configurar o widget visão geral do trigger **Trigger overview** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos do widget `fields` não são validadas durante a criação ou atualização de um painel. Isso permite que os usuários modifiquem os widgets integrados **built-in widgets** e criem widget personalizados **custom widgets**, mas também introduz o risco de criar e atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget **Trigger overview**, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget **Trigger Overview** (visão geral do trigger).

Parâmetro	type	nome	valor
Intervalo de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Mostrar	0	show	1 - (padrão) Problema recentes; 2 - Qualquer; 3 - Problemas.
Grupos de host	2	groupids	Host group ID. Observe: Para configurar vários grupos de hosts, crie um objeto de campo de widget de painel para cada grupo de hosts.
Hosts	3	hostids	Host ID. Observe: Para configurar vários hosts, crie um objeto de campo de widget de painel para cada host. Para vários hosts, o parâmetro Grupo de hosts deve ser configurado com pelo menos um grupo de hosts ao qual os hosts configurados pertencem.

Parâmetro	type	nome	valor
Tags (o número no nome da pro- priedade (por ex- em- plo, tags.tag.0) refere- se à or- dem da tag na lista de avali- ação de tags)			
Tipo de avaliação	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Nome da tag	1	tags.tag.0	Qualquer valor da string.
Operador	0	tags.operator.0	Parâmetro Nome da tag obrigatório ao configurar Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
Valor da tag	1	tags.value.0	Parâmetro Operador obrigatório ao configurar Tags. Qualquer valor da tag.
Mostrar prob- le- mas suprim- i- dos	0	show_suppressed	Parâmetro Valor da tag obrigatório ao configurar Tags. 0 - (default) Desativado; 1 - Ativado.
Localização dos hosts	0	style	0 - (padrão) Esquerda; 1 - Superior.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo de widget do painel para o widget Trigger overview (visão geral do trigger). Para mais informações sobre como configurar um painel, consulte [dashboard.create](#).

Configurando um widget de Visão geral do Trigger

Configure um widget de * Visão geral do Trigger* que exiba os estados dos triggers para todos os grupos de hosts que possuem

gatilhos com uma tag que tenha o nome "escopo" e contenha o valor "disponibilidade".

that displays trigger states for all host groups that have triggers with a tag that has the name "scope" and contains value "availability".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "trigover",
            "name": "Trigger overview",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 1,
                "name": "tags.tag.0",
                "value": "scope"
              },
              {
                "type": 0,
                "name": "tags.operator.0",
                "value": 0
              },
              {
                "type": 1,
                "name": "tags.value.0",
                "value": "availability"
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
    {
      "usrgrpid": 7,
      "permission": 2
    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}
```

Consulte também

-[Campo widget do dashboard](#) -[dashboard.create](#) -[dashboard.update](#)

23 URL

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo de widget de painel respectivos permitem configurar o widget [URL](#) nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos `fields` do widget não são validadas durante a criação ou atualização de um painel. Isso permite que os usuários modifiquem os widgets integrados [built-in widgets](#) e criem os widgets personalizados [custom widgets](#), mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget URL, consulte os requisitos de parâmetros delineados nas tabelas abaixo.

Parâmetros

Os seguintes parâmetros são suportados para o widget URL.

Parâmetro	type	nome	valor
Intervalo de atualização	0	<code>rf_rate</code>	0 - (padrão) Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
URL (obrigatório)	1	<code>url</code>	String de URL válida URL.
Item dinâmico	0	<code>dynamic</code>	0 - (padrão) Desativado; 1 - Ativado.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo do widget de painel para URL. Para mais informações sobre como configurar um dashboard, consulte o manual em [dashboard.create](#).

Configurando um URL*

Configure um widget URL que exibe a página inicial do manual do Zabbix.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
```

```

{
  "widgets": [
    {
      "type": "url",
      "name": "URL",
      "x": 0,
      "y": 0,
      "width": 12,
      "height": 5,
      "view_mode": 0,
      "fields": [
        {
          "type": 1,
          "name": "url",
          "value": "https://www.zabbix.com/documentation/6.0/en"
        }
      ]
    }
  ],
  "userGroups": [
    {
      "usrgrpid": 7,
      "permission": 2
    }
  ],
  "users": [
    {
      "userid": 1,
      "permission": 3
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

24 Monitoramento Web

Descrição

Esses parâmetros e os possíveis valores de propriedade para os objetos de campo do widget de painel de controle permitem configurar o widget de **Web monitoring** nos métodos `dashboard.create` e `dashboard.update`.

Attention:

As propriedades dos campos do widget `fields` não são validadas durante a criação ou atualização de um painel de controle. Isso permite aos usuários modificar os **built-in widgets** e criar **custom widgets**, mas também introduz o risco de criar ou atualizar widgets incorretamente. Para garantir a criação ou atualização bem-sucedida do widget Monitoramento web, consulte os requisitos de parâmetro delineados nas tabelas abaixo.

Parâmetros

Os parâmetros a seguir são suportados pelo widget Monitoramento Web

Parâmetro	tipo	nome	valor
Valor de atualização	0	rf_rate	0 - Sem atualização; 10 - 10 segundos; 30 - 30 segundos; 60 - (padrão) 1 minuto; 120 - 2 minutos; 600 - 10 minutos; 900 - 15 minutos.
Grupos de Host	2	groupids	Host group ID. Note: Para configurar vários grupos de hosts, crie um objeto de campo de widget de painel para cada grupo de hosts.
Excluir grupos de hosts	2	exclude_groupids	Host group ID. Note: Para excluir vários grupos de hosts, crie um objeto de campo de widget de painel para cada grupo de hosts.
Hosts	3	hostids	ID do Host . Note: Para configurar vários hosts, crie um objeto de campo de widget de painel para cada host. Para vários hosts, o parâmetro Grupos de hosts deve ser não configurado ou configurado com pelo menos um grupo de host ao qual os hosts configurados pertencem.

Tags
(o
número
no
nome
da
pro-
priedade
(por
ex-
em-
plo,
tags.tag.0)
faz
refer-
ên-
cia
à
or-
dem
das
tags
na
lista
de
avali-
ação
de
tags)

Parâmetro	tipo	nome	valor
Tipo de avaliação	0	evaltype	0 - (padrão) E/Ou; 2 - Ou.
Nome da tag	1	tags.tag.0	Qualquer valor de string.
Operador	0	tags.operator.0	Parâmetro Nome da tag obrigatório se configurando Tags. 0 - Contém; 1 - Igual; 2 - Não contém; 3 - Diferente de; 4 - Existe; 5 - Não existe.
Valor da tag	1	tags.value.0	Parâmetro Operador obrigatório se configurando Tags. Qualquer valor de string.
Mostrar hosts em manutenção	0	maintenance	Parâmetro Valor da tag obrigatório se configurando Tags. 0 - Desabilitado; 1 - (padrão) Habilitado.

Exemplos

Os seguintes exemplos têm como objetivo apenas descrever a configuração dos objetos de campo de widget do painel para o widget de Monitoramento da Web. Para obter mais informações sobre como configurar um painel, consulte [dashboard.create](#).

Configurando um widget de Monitoramento Web

Configure um widget de Monitoramento Web que exibe um resumo do status dos cenários de monitoramento web ativos para o grupo de hosts "4".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "dashboard.create",
  "params": {
    "name": "My dashboard",
    "display_period": 30,
    "auto_start": 1,
    "pages": [
      {
        "widgets": [
          {
            "type": "web",
            "name": "Web monitoring",
            "x": 0,
            "y": 0,
            "width": 6,
            "height": 3,
            "view_mode": 0,
            "fields": [
              {
                "type": 2,
                "name": "groupids",
                "value": 4
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
```



```

    {
      "usrgrpid": 7,
      "permission": 2
    },
    "users": [
      {
        "userid": 1,
        "permission": 3
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "3"
    ]
  },
  "id": 1
}

```

Consulte também

- [Dashboard widget field](#)
- [dashboard.create](#)
- [dashboard.update](#)

painel.atualização

Descrição

painel de objetos.update(painéis de objetos/matrizes)

Este método permite atualizar os painéis existentes.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do painel a serem atualizadas.

A propriedade dashboardid deve ser especificada para cada painel, todas outras propriedades são opcionais. Somente as propriedades especificadas serão Atualizada.

Além das [propriedades padrão do painel](#), o método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [pages|array|Painel [páginas](#) para substituir as páginas do painel existentes.

As páginas do painel são atualizadas pela propriedade dashboard_pageid. Novas páginas de painel serão criadas para objetos sem a propriedade dashboard_pageid e as páginas de painel existentes serão excluídas se não forem reutilizadas. As páginas do painel serão ordenadas na mesma ordem especificada. Somente as propriedades especificadas das páginas do painel serão atualizadas. Pelo menos um objeto de página do painel é necessário para a propriedade pages.| [users|array|Dashboard [user](#) compartilha para substituir os elementos existentes.| [userGroups|array|Dashboard [grupo de usuários](#) compartilha para substituir os elementos existentes.]

Valores de retorno

(object) Retorna um objeto contendo os IDs do painéis sob a propriedade dashboardids.

Exemplos

Renomeando um painel

Renomeie um painel para "status do servidor SQL".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dashboard.update",
  "parâmetros": {
    "dashboardid": "2",
    "name": "status do servidor SQL"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "dashboardids": [
      "2"
    ]
  },
  "id": 1
}
```

Atualizando páginas do painel

Renomeie a primeira página do painel, substitua os widgets no segundo painel página e adicione uma nova página como a terceira. Excluir todos os outros painéis Páginas.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dashboard.update",
  "parâmetros": {
    "dashboardid": "2",
    "Páginas": [
      {
        "dashboard_pageid": 1,
        "name": 'Página renomeada'
      },
      {
        "dashboard_pageid": 2,
        "widgets": [
          {
            "tipo": "relógio",
            "x": 0,
            "y": 0,
            "largura": 4,
            "altura": 3
          }
        ]
      }
    ],
    {
      "display_period": 60
    }
  ]
},
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "dashboardids": [
      "2"
    ]
  },
  "id": 2
}
```

Alterar o proprietário do painel

Disponível apenas para administradores e superadministradores.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dashboard.update",
  "parâmetros": {
    "dashboardid": "2",
    "userid": "1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 2
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "dashboardids": [
      "2"
    ]
  },
  "id": 2
}
```

Veja também

- [Página do painel](#)
- [Widget do painel](#)
- [Campo de widget do painel](#)
- [usuário do painel](#)
- [grupo de usuários do painel](#)

Fonte

CDashboard::update() em ui/include/classes/api/services/CDashboard.php.

painel.criar

Descrição

painel de objetos.create(painéis de objetos/matriz)

Este método permite criar novos dashboards.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Dashboards para criar.

Além das [propriedades padrão do painel](#), o método aceita os seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| | **páginas**
(obrigatório)|array|Painel **páginas** a ser criado para o painel. As páginas do painel serão ordenadas na mesma ordem especificada. Pelo menos um objeto de página do painel é necessário para a propriedade pages.| [users|array|Dashboard **user** compartilhamentos a serem criados no painel.| [userGroups|array|Dashboard **grupo de usuários** compartilhamentos a serem criados no painel.]

Valores de retorno

(object) Retorna um objeto contendo os IDs do criado painéis sob a propriedade dashboardids. A ordem do retorno Os IDs correspondem à ordem dos painéis passados.

Exemplos

Criando um painel

Crie um painel chamado "Meu painel" com um widget de Problemas com tags e usando dois tipos de compartilhamento (grupo de usuários e usuário) em um único página do painel.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dashboard.create",
  "parâmetros": {
    "name": "Meu painel",
    "display_period": 30,
    "auto_start": 1,
    "Páginas": [
      {
        "widgets": [
          {
            "tipo": "problemas",
            "x": 0,
            "y": 0,
            "largura": 12,
            "altura": 5,
            "view_mode": 0,
            "Campos": [
              {
                "tipo": 1,
                "nome": "tags.tag.0",
                "valor": "serviço"
              },
              {
                "tipo": 0,
                "nome": "tags.operador.0",
                "valor": 1
              },
              {
                "tipo": 1,
                "nome": "tags.value.0",
                "valor": "zabbix_server"
              }
            ]
          }
        ]
      }
    ]
  },
  "userGroups": [
    {
      "usrgrpId": "7",
      "permissão": 2
    }
  ],
  "Comercial": [
    {
      "userid": "4",
```

```

        "permissão": 3
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 1
}

```

Veja também

- [Página do painel](#)
- [Widget do painel](#)
- [Campo de widget do painel](#)
- [usuário do painel](#)
- [grupo de usuários do painel](#)

Fonte

CDashboard::create() em ui/include/classes/api/services/CDashboard.php.

painel.delete

Descrição

objeto dashboard.delete(array dashboardids)

Este método permite excluir painéis.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos painéis a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs do painéis sob a propriedade dashboardids.

Exemplos

Excluindo vários painéis

Exclua dois painéis.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "dashboard.delete",
    "parâmetros": [
        "2",
        "3"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
}

```

```
    "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "dashboardids": [
      "2",
      "3"
    ]
  },
  "id": 1
}
```

Fonte

CDashboard::delete() in ui/include/classes/api/services/CDashboard.php.

#####painel.get {#manual-api-reference-dashboard-get}

Descrição

integer/array dashboard.get(object parameters)

O método permite recuperar dashboards de acordo com os dados parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [dashboardids|string/array|Retorne apenas painéis com os IDs fornecidos.] [selectPages|query|Retorne uma propriedade **pages** com páginas do painel, ordenadas corretamente.] [selectUsers|query|Retorne uma propriedade **users** com os usuários com os quais o painel é compartilhado.] [selectUserGroups|query|Retorne uma propriedade **userGroups** com grupos de usuários com os quais o painel é compartilhado.] [sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

O valor possível é: dashboardid.] [countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página **comentário de referência**.] [editável|booleano|^] [excludePesquisa|boolean|^] [filtro|objeto|^] [limite|inteiro|^] [saída|consulta|^] [preservekeys|boolean|^] [pesquisar|objeto|^] [searchByAny|boolean|^] [searchWildcardsEnabled|boolean|^] [ordem de classificação|cadeia/matriz|^] [startSearch|boolean|^]

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando um painel por ID

Recupere todos os dados sobre os painéis "1" e "2".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dashboard.get",
  "parâmetros": {
    "saída": "estender",
    "selectPages": "estender",
    "selectUsers": "estender",
  }
}
```

```

        "selectUserGroups": "estender",
        "dashboardids": [
            "1",
            "2"
        ]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": [
        {
            "dashboardid": "1",
            "nome": "Painel",
            "userid": "1",
            "privado": "0",
            "display_period": "30",
            "auto_start": "1",
            "Comercial": [],
            "userGroups": [],
            "Páginas": [
                {
                    "dashboard_pageid": "1",
                    "nome": "",
                    "display_period": "0",
                    "widgets": [
                        {
                            "widgetid": "9",
                            "tipo": "informações do sistema",
                            "nome": "",
                            "x": "12",
                            "y": "8",
                            "largura": "12",
                            "altura": "5",
                            "view_mode": "0",
                            "Campos": []
                        },
                        {
                            "widgetid": "8",
                            "type": "problemsbysv",
                            "nome": "",
                            "x": "12",
                            "s": "4",
                            "largura": "12",
                            "altura": "4",
                            "view_mode": "0",
                            "Campos": []
                        },
                        {
                            "widgetid": "7",
                            "type": "hosts do problema",
                            "nome": "",
                            "x": "12",
                            "y": "0",
                            "largura": "12",
                            "altura": "4",
                            "view_mode": "0",
                            "Campos": []
                        }
                    ]
                }
            ]
        }
    ]
}

```

```

    {
      "widgetid": "6",
      "tipo": "descoberta",
      "nome": "",
      "x": "6",
      "s": "9",
      "largura": "6",
      "altura": "4",
      "view_mode": "0",
      "Campos": []
    },
    {
      "widgetid": "5",
      "tipo": "web",
      "nome": "",
      "x": "0",
      "s": "9",
      "largura": "6",
      "altura": "4",
      "view_mode": "0",
      "Campos": []
    },
    {
      "widgetid": "4",
      "tipo": "problemas",
      "nome": "",
      "x": "0",
      "s": "3",
      "largura": "12",
      "altura": "6",
      "view_mode": "0",
      "Campos": []
    },
    {
      "widgetid": "3",
      "type": "favmaps",
      "nome": "",
      "x": "8",
      "y": "0",
      "largura": "4",
      "altura": "3",
      "view_mode": "0",
      "Campos": []
    },
    {
      "widgetid": "1",
      "type": "favgraphs",
      "nome": "",
      "x": "0",
      "y": "0",
      "largura": "4",
      "altura": "3",
      "view_mode": "0",
      "Campos": []
    }
  ]
},
{
  "dashboard_pageid": "2",
  "nome": "",
  "display_period": "0",
  "widgets": []
}

```



```

    },
    {
        "dashboard_pageid": "3",
        "name": "Nome da página personalizada",
        "display_period": "60",
        "widgets": []
    }
]
},
{
    "dashboardid": "2",
    "name": "Meu painel",
    "userid": "1",
    "privado": "1",
    "display_period": "60",
    "auto_start": "1",
    "Comercial": [
        {
            "userid": "4",
            "permissão": "3"
        }
    ],
    "userGroups": [
        {
            "usrgrpid": "7",
            "permissão": "2"
        }
    ],
    "Páginas": [
        {
            "dashboard_pageid": "4",
            "nome": "",
            "display_period": "0",
            "widgets": [
                {
                    "widgetid": "10",
                    "tipo": "problemas",
                    "nome": "",
                    "x": "0",
                    "y": "0",
                    "largura": "12",
                    "altura": "5",
                    "view_mode": "0",
                    "Campos": [
                        {
                            "tipo 2",
                            "nome": "grupoids",
                            "valor": "4"
                        }
                    ]
                }
            ]
        }
    ]
},
],
    "id": 1
}

```

Veja também

- [Página do painel](#)
- [Widget do painel](#)

- [Campo de widget do painel](#)
- [usuário do painel](#)
- [grupo de usuários do painel](#)

Fonte

CDashboard::get() em ui/include/classes/api/services/CDashboard.php.

Painel de Modelos

Esta classe foi projetada para trabalhar com painéis de modelos.

Object references:

- [Painel de Modelos](#)
- [Página do Painel de Modelos](#)
- [Widget Painel de Modelos](#)
- [Campo de Painel de Modelos](#)

Métodos Disponíveis:

- [templatedashboard.create](#) - creating new template dashboards
- [templatedashboard.delete](#) - deleting template dashboards
- [templatedashboard.get](#) - retrieving template dashboards
- [templatedashboard.update](#) - updating template dashboards

> Objeto de dashboard

Os objetos a seguir estão diretamente relacionados à API `templatedashboard`.

Template dashboard

The template dashboard object has the following properties.

Property	Type	Description
dashboardid	string	(readonly) ID of the template dashboard.

O objeto de painel de modelo tem as seguintes propriedades.

Propriedade	Type	Descrição
dashboardid	string	(somente leitura) ID do template do dashboard.
name (requerido)	string	Nome do template do dashboard.
templateid (requerido)	string	ID do modelo ao qual o dashboard pertence.
display_period	integer	Período de exibição da página padrão (em segundos). Valores possíveis: 10, 30, 60, 120, 600, 1800, 3600. Padrão: 30.
auto_start	integer	Iniciar apresentação de slides automaticamente. Valores possíveis: 0 - não iniciar apresentação de slides automaticamente; 1 - (padrão) iniciar apresentação de slides automaticamente.
uuid	string	Identificador exclusivo universal, usado para vincular painéis de modelos importados aos já existentes. Gerado automaticamente, se não fornecido.
name (required)	string	Para operações de atualização, este campo é somente leitura. Name of the template dashboard.

Propriedade	Type	Descrição
templateid (required)	string	ID of the template the dashboard belongs to.
display_period	integer	Default page display period (in seconds). Possible values: 10, 30, 60, 120, 600, 1800, 3600. Default: 30.
auto_start	integer	Auto start slideshow. Possible values: 0 - do not auto start slideshow; 1 - (default) auto start slideshow.
uuid	string	Universal unique identifier, used for linking imported template dashboards to already existing ones. Auto-generated, if not given. For update operations this field is readonly.

Página dos modelos de dashboard

O objeto de página de modelo de dashboard tem as seguintes propriedades.

Propriedade	Type	Descrição
dashboard_pageid	string	(somente leitura) ID da dashboard do painel.
name	string	Nome da página do painel. Padrão: string vazia.
display_period	integer	Período de exibição da página do painel (em segundos). Valores possíveis: 0, 10, 30, 60, 120, 600, 1800, 3600.
widgets	array	Padrão: 0 (usará a exibição de página padrão período). Array dos objetos template dashboard widget .

Widget do modelo de dashboard

O objeto de widget de modelo de dashboard tem as seguintes propriedades.

Propriedade	Type	Descrição
widgetid	string	(somente leitura) ID do widget do dashboard.
type (requerido)	string	Tipo de widget do dashboard. Valores possíveis: clock - Relógio; graph - Gráfico (clássico); graphprototype - Protótipo de gráfico; item - Valor do item; plaintext - Texto simples; url - URL;
name	string	Nome do widget personalizado.
x	integer	Uma posição horizontal do lado esquerdo do painel. Os valores válidos variam de 0 a 23.
y	integer	Uma posição vertical na parte superior do painel. Os valores válidos variam de 0 a 62.
width	integer	A largura do widget. Os valores válidos variam de 1 a 24.

Propriedade	Type	Descrição
height	integer	A altura do widget.
view_mode	integer	Os valores válidos variam de 2 a 32. O modo de visualização de widget.
fields	array	Valores possíveis: 0 - (padrão) visualização de widget padrão; 1 - com cabeçalho oculto; Array dos objetos campo de widget do painel de modelo .

Campo de widget de modelo de dashboard

O objeto de campo do widget de modelo de dashboard tem as seguintes propriedades.

Propriedade	Type	Descrição
type (requerido)	integer	Tipo do campo do widget. Valores possíveis: 0 - Integer; 1 - String; 4 - Item; 5 - Protótipo do item; 6 - Gráfico ; 7 - Protótipo de gráfico.
nome	string	Nome do campo do widget.
value (requerido)	mixed	Valor do campo do widget dependendo do tipo.

templatedashboard.create

Descrição

`object templatedashboard.create(object/array templateDashboards)`

Este método permite criar novos painéis de modelo.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte **Funções do usuário** para obter mais informações.

Parâmetros

(object/array) Painéis de modelo a serem criados.

Além das **propriedades do painel de modelo padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
páginas (obrigatório)	array	Páginas de painel de modelo a serem criadas para o painel. As páginas do painel serão ordenadas na mesma ordem específica. Pelo menos um objeto de página do painel é necessário para a página propriedade.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos painéis de modelo criados na propriedade `dashboardids`. A ordem dos IDs retornados corresponde à ordem dos painéis de modelo passados

Exemplos

Criando um template de dashboard

Crie um modelo de dashboard chamado "Gráficos" com um widget de gráfico em uma única página de dashboard.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "templatedashboard.create",
  "params": {
    "templateid": "10318",
    "name": "Graphs",
    "pages": [
      {
        "widgets": [
          {
            "type": "graph",
            "x": 0,
            "y": 0,
            "width": 12,
            "height": 5,
            "view_mode": 0,
            "fields": [
              {
                "type": 6,
                "name": "graphid",
                "value": "1123"
              }
            ]
          }
        ]
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "32"
    ]
  },
  "id": 1
}
```

Veja também

- [Página do painel de modelos](#)
- [Widget do painel de modelos](#)
- [Campo do widget do painel de modelos](#)

Fonte

CTemplateDashboard::create() in ui/include/classes/api/services/CTemplateDashboard.php.

templatedashboard.delete

Descrição

object templatedashboard.delete(array templateDashboardIds)

Este método permite excluir painéis de modelo.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs de templates de dashboard para deletar.

Retorno de valores

(object) Retorna um objeto contendo os IDs dos templates de dashboards deletados na propriedade `dashboardids`.

Exemplos

Deletando vários templates de dashboards

Deleta dois templates de dashboards.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "templatedashboard.delete",
  "params": [
    "45",
    "46"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "45",
      "46"
    ]
  },
  "id": 1
}
```

Fonte

CTemplateDashboard::delete() in ui/include/classes/api/services/CTemplateDashboard.php.

templatedashboard.get**Descrição**

integer/array templatedashboard.get(object parameters)

O método permite recuperar painéis de modelo de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
dashboardids	string/array	Retorne apenas painéis de modelo com os IDs fornecidos.

Parâmetro	Tipo	Descrição
templateids	string/array	Retorne apenas os painéis de modelo que pertencem aos modelos fornecidos.
selectPages	query	Retorne uma propriedade de páginas com páginas de painel de modelo, ordenadas corretamente.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: dashboardid and name. Esses parâmetros sendo comuns para todos os get métodos são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(inteiro/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando painéis de modelo

Recupere todos os painéis de modelo com widgets para um modelo especificado. Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "templatedashboard.get",
  "params": {
    "output": "extend",
    "selectPages": "extend",
    "templateids": "10001"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "dashboardid": "23",
      "name": "Docker overview",
      "templateid": "10001",
      "display_period": "30",
      "auto_start": "1",
      "uuid": "6dfcbe0bc5ad400ea9c1c2dd7649282f",
      "pages": [
        {
          "dashboard_pageid": "1",
          "name": "",
          "display_period": "0",
          "widgets": [
```

```

{
  "widgetid": "220",
  "type": "graph",
  "name": "",
  "x": "0",
  "y": "0",
  "width": "12",
  "height": "5",
  "view_mode": "0",
  "fields": [
    {
      "type": "6",
      "name": "graphid",
      "value": "1125"
    }
  ]
},
{
  "widgetid": "221",
  "type": "graph",
  "name": "",
  "x": "12",
  "y": "0",
  "width": "12",
  "height": "5",
  "view_mode": "0",
  "fields": [
    {
      "type": "6",
      "name": "graphid",
      "value": "1129"
    }
  ]
},
{
  "widgetid": "222",
  "type": "graph",
  "name": "",
  "x": "0",
  "y": "5",
  "width": "12",
  "height": "5",
  "view_mode": "0",
  "fields": [
    {
      "type": "6",
      "name": "graphid",
      "value": "1128"
    }
  ]
},
{
  "widgetid": "223",
  "type": "graph",
  "name": "",
  "x": "12",
  "y": "5",
  "width": "12",
  "height": "5",
  "view_mode": "0",
  "fields": [
    {

```



```

        "type": "6",
        "name": "graphid",
        "value": "1126"
    }
    ],
    },
    {
        "widgetid": "224",
        "type": "graph",
        "name": "",
        "x": "0",
        "y": "10",
        "width": "12",
        "height": "5",
        "view_mode": "0",
        "fields": [
            {
                "type": "6",
                "name": "graphid",
                "value": "1127"
            }
        ]
    }
    ],
    },
    ],
    "id": 1
}

```

Veja também

- [Página de template para dashboards](#)
- [Widget de template para dashboards](#)
- [Campo widget para template dashboard](#)

Fonte

CTemplateDashboard::get() in ui/include/classes/api/services/CTemplateDashboard.php.

templatedashboard.update

Descrição

object templatedashboard.update(object/array templateDashboards)

Este método permite atualizar templates de dashboards já existentes.

Note:

Disponível somente para os perfis Admin e Super admin. Permissões para utilizar o método podem ser revogadas nas propriedades do usuário. Veja [User roles](#) para mas informações.

Parâmetros

(object/array) Propriedades do painel do modelo a serem atualizadas.

A propriedade dashboardid deve ser especificada para cada painel, todas as outras propriedades são opcionais. Somente as propriedades especificadas serão atualizadas.

Além das [propriedades do painel de modelo padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
páginas	array	<p>páginas de painel de modelo para substituir as páginas de painel existentes.</p> <p>As páginas do painel são atualizadas pela propriedade <code>dashboard_pageid</code>. Novas páginas de painel serão criadas para objetos sem a propriedade <code>dashboard_pageid</code> e as páginas de painel existentes serão excluídas se não forem reutilizadas. As páginas do painel serão ordenadas na mesma ordem especificada. Somente as propriedades especificadas das páginas do painel serão atualizadas. Pelo menos um objeto de página do painel é necessário para a propriedade <code>pages</code>.</p>

Retorno de valores

(object) Retorna um objeto contendo os IDs dos templates de dashboards atualizados na propriedade `dashboardids`.

Exemplos

Renomeando um template de dashboard

Renomeia um template de dashboard para "Performance graphs".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "templatedashboard.update",
  "params": {
    "dashboardid": "23",
    "name": "Performance graphs"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "dashboardids": [
      "23"
    ]
  },
  "id": 1
}
```

Atualizando páginas de templates de dashboard

Renomeia a primeira página de dashboard, substitui widgets na segunda página de dashboard e adiciona uma terceira nova página. Deleta todas as outras páginas de dashboard.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "templatedashboard.update",
  "params": {
    "dashboardid": "2",
    "pages": [
      {
        "dashboard_pageid": 1,
        "name": 'Renamed Page'
      },
      {
        "dashboard_pageid": 2,
```

```

        "widgets": [
            {
                "type": "clock",
                "x": 0,
                "y": 0,
                "width": 4,
                "height": 3
            }
        ],
        {
            "display_period": 60
        }
    ],
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "result": {
        "dashboardids": [
            "2"
        ]
    },
    "id": 2
}

```

Veja também

- [Template dashboard widget](#)
- [Template dashboard widget field](#)

Source

CTemplateDashboard::update() in ui/include/classes/api/services/CTemplateDashboard.php.

Problema

Esta classe é projetada para trabalhar com problemas.

Referências de objetos:

- [Problema](#)

Métodos disponíveis:

- [problem.get](#) - recuperando problemas

> Objeto do problema

Note:

problemas são criados pelo servidor Zabbix e não podem ser modificados via API.

O objeto de problema tem as seguintes propriedades.

Propriedade	Tipo	Descrição
eventid	string	ID do evento do problema.

Propriedade	Tipo	Descrição
source	integer	Tipo do evento do problema. Valores possíveis: 0 - evento criado por uma trigger; 3 - evento interno; 4 - evento criado na atualização do status do serviço.
object	integer	Tipo de objeto relacionado ao evento do problema. Possíveis valores para triggers de eventos: 0 - trigger. Valores possíveis para eventos internos: 0 - trigger; 4 - item; 5 - Regra LLD. Valores possíveis para eventos de serviço: 6 - serviço.
objectid	string	ID do objeto relacionado.
clock	timestamp	Hora em que o evento do problema foi criado.
ns	integer	Nanossegundos quando o evento do problema foi criado.
r_eventid	string	ID do evento de recuperação.
r_clock	timestamp	Hora em que o evento de recuperação foi criado.
r_ns	integer	Nanossegundos quando o evento de recuperação foi criado.
correlationid	string	ID da regra de correlação se este evento foi recuperado pela regra de correlação global.
userid	string	ID do usuário se o problema foi fechado manualmente.
name	string	Nome do problema resolvido.
acknowledged	integer	Reconheça o estado do problema. Valores possíveis: 0 - não reconhecido; 1 - reconhecido.
severity	integer	Gravidade atual do problema. Valores possíveis: 0 - não classificado; 1 - informações; 2 - aviso; 3 - médio; 4 - alto; 5 - desastre.
suppressed	integer	Se o problema foi suprimido. Valores possíveis: 0 - o problema está em estado normal; 1 - o problema foi suprimido.
opdata	string	Dados operacionais com macros expandidas.
urls	array de URLs de tipo de mídia	URLs de tipos de mídia ativos.

Problem

Note:

Problems are created by the Zabbix server and cannot be modified via the API.

The problem object has the following properties.

Property	Type	Description
eventid	string	ID of the problem event.
source	integer	Type of the problem event. Possible values: 0 - event created by a trigger; 3 - internal event; 4 - event created on service status update.
object	integer	Type of object that is related to the problem event. Possible values for trigger events: 0 - trigger. Possible values for internal events: 0 - trigger; 4 - item; 5 - LLD rule. Possible values for service events: 6 - service.
objectid	string	ID of the related object.
clock	timestamp	Time when the problem event was created.
ns	integer	Nanoseconds when the problem event was created.
r_eventid	string	Recovery event ID.
r_clock	timestamp	Time when the recovery event was created.
r_ns	integer	Nanoseconds when the recovery event was created.
correlationid	string	Correlation rule ID if this event was recovered by global correlation rule.
userid	string	User ID if the problem was manually closed.
name	string	Resolved problem name.
acknowledged	integer	Acknowledge state for problem. Possible values: 0 - not acknowledged; 1 - acknowledged.
severity	integer	Problem current severity. Possible values: 0 - not classified; 1 - information; 2 - warning; 3 - average; 4 - high; 5 - disaster.
suppressed	integer	Whether the problem is suppressed. Possible values: 0 - problem is in normal state; 1 - problem is suppressed.
opdata	string	Operational data with expanded macros.
urls	array of Media type URLs	Active media types URLs.

Etiqueta do problema

O objeto de tag de problema tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tag	string	Nome da tag do problema.
value	string	Valor da etiqueta do problema.

URLs de tipo de mídia

Objeto com URL de tipo de mídia tem as seguintes propriedades.

Propriedade	Tipo	Descrição
name	string	Nome do URL definido pelo tipo de mídia.
url	string	Valor de URL definido por tipo de mídia.

Os resultados conterão entradas apenas para tipos de mídia ativos com entrada de menu de eventos ativada. A macro usada nas propriedades será expandida, mas se uma das propriedades contiver macro não expandida, ambas as propriedades serão excluídas dos resultados. Macros compatíveis descritas em [página](/manual/appendix/macros/supported_by_location

problem.get

Descrição

integer/array problem.get(object parameters)

O método permite recuperar problemas de acordo com os parâmetros fornecidos.

Este método é para recuperar problemas não resolvidos. Também é possível, se especificado, recuperar problemas resolvidos recentemente. O período que determina a idade "recentemente" é definido em Administração → Geral.

Os problemas que foram resolvidos antes desse período não são mantidos na tabela de problemas. Para recuperar problemas que foram resolvidos no passado, use o método [event.get](#).

Attention:

Esse método pode retornar problemas de uma entidade excluída se esses problemas ainda não tiverem sido removidos pelo housekeeper .

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
eventids	string/array	Retornar apenas problemas com os IDs fornecidos.
groupids	string/array	Retorna apenas problemas criados por objetos que pertencem aos grupos de hosts fornecidos.
hostids	string/array	Retorna apenas problemas criados por objetos que pertencem aos hosts fornecidos.
objectids	string/array	Retorna apenas os problemas criados pelos objetos fornecidos.
source	integer	Retorna apenas problemas com o tipo fornecido.
		Consulte a página de objeto de evento de problema para obter uma lista de tipos de eventos compatíveis.
object	integer	Padrão: 0 - problema criado por uma trigger. Retorne apenas os problemas criados por objetos de um determinado tipo.
		Consulte a página do objeto de evento do problema para obter uma lista de tipos de objetos compatíveis.
acknowledged	boolean	Padrão: 0 - trigger. true - retornar somente problemas reconhecidos; false - somente não reconhecidos.

Parâmetro	Tipo	Descrição
suppressed	boolean	<p><code>true</code> - retorna apenas problemas suprimidos; <code>false</code> - retorna problemas no estado normal.</p>
severities	integer/array	<p>Retorne apenas problemas com determinadas gravidades de eventos. Aplica-se apenas se o objeto for trigger.</p>
evaltype	integer	<p>Regras para pesquisa de tags.</p>
tags	array of objects	<p>Valores possíveis: 0 - (padrão) And/Or; 2 - Or.</p> <p>Retorna apenas problemas com as tags fornecidas. Correspondência exata por tag e pesquisa sem distinção entre maiúsculas e minúsculas por valor e operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Um array vazio retorna todos os problemas.</p> <p>Tipos de operadores possíveis: 0 - (padrão) Like; 1 - Equal; 2 - Not like; 3 - Not equal 4 - Exists; 5 - Not exists.</p>
recent	boolean	<p><code>true</code> - Retorna PROBLEM e recentemente RESOLVED problemas (depende dos acionadores de exibição OK por N segundos) Padrão: <code>false</code> - apenas problemas UNRESOLVED</p>
eventid_from	string	Retornar apenas problemas com IDs maiores ou iguais ao ID fornecido.
eventid_till	string	Retornar apenas problemas com IDs menores ou iguais ao ID fornecido.
time_from	timestamp	Retornar apenas os problemas que foram criados após ou no momento determinado.
time_till	timestamp	Retornar apenas problemas que foram criados antes ou no momento determinado.
selectAcknowledges	query	<p>Retorna uma propriedade <code>acknowledges</code> com as atualizações do problema. As atualizações de problemas são classificadas em ordem cronológica inversa.</p> <p>O objeto de atualização do problema tem as seguintes propriedades: <code>acknowledgeid</code> - (string) ID da atualização; <code>userid</code> - (string) ID do usuário que atualizou o evento; <code>eventid</code> - (string) ID do evento atualizado; <code>clock</code> - (timestamp) hora em que o evento foi atualizado; <code>message</code> - (string) texto da mensagem; <code>action</code> - (integer) tipo de ação de atualização (veja event.acknowledge); <code>old_severity</code> - (integer) gravidade do evento antes desta ação de atualização; <code>new_severity</code> - (integer) gravidade do evento após esta ação de atualização;</p>
selectTags	query	<p>Suporta count.</p> <p>Retorne uma propriedade <code>tags</code> com as tags de problema. Formato de saída: [{"tag": "<tag>", "value": "<value>"}, ...].</p>
selectSuppressionData	query	<p>Retorna uma propriedade <code>suppression_data</code> com a lista de manutenções: <code>maintenanceid</code> - (string) ID da manutenção; <code>suppress_until</code> - (integer) tempo até que o problema seja suprimido.</p>
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	<p>Os valores possíveis são: <code>eventid</code>. Esses parâmetros sendo comuns para todos os métodos <code>get</code> são descritos em detalhes na página comentário de referência page.</p>

Parâmetro	Tipo	Descrição
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando eventos de problema de trigger

Recupere eventos recentes do trigger "15112".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "problem.get",
  "params": {
    "output": "extend",
    "selectAcknowledges": "extend",
    "selectTags": "extend",
    "selectSuppressionData": "extend",
    "objectids": "15112",
    "recent": "true",
    "sortfield": ["eventid"],
    "sortorder": "DESC"
  },
  "auth": "67f45d3eb1173338e1b1647c4bdc1916",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "eventid": "1245463",
      "source": "0",
      "object": "0",
      "objectid": "15112",
      "clock": "1472457242",
      "ns": "209442442",
      "r_eventid": "1245468",
      "r_clock": "1472457285",
      "r_ns": "125644870",
      "correlationid": "0",
      "userid": "1",
      "name": "Zabbix agent on localhost is unreachable for 5 minutes",
      "acknowledged": "1",
      "severity": "3",

```



```

        "opdata": "",
        "acknowledges": [
            {
                "acknowledgeid": "14443",
                "userid": "1",
                "eventid": "1245463",
                "clock": "1472457281",
                "message": "problem solved",
                "action": "6",
                "old_severity": "0",
                "new_severity": "0"
            }
        ],
        "suppression_data": [
            {
                "maintenanceid": "15",
                "suppress_until": "1472511600"
            }
        ],
        "suppressed": "1",
        "tags": [
            {
                "tag": "test tag",
                "value": "test value"
            }
        ]
    },
    "id": 1
}

```

Veja também

- [Alert](#)
- [Item](#)
- [Host](#)
- [LLD rule](#)
- [Trigger](#)

Fonte

CEvent::get() in ui/include/classes/api/services/CProblem.php.

Protótipo de gráfico

Esta classe é projetada para trabalhar com protótipos de gráficos.

Referências de objetos:

- [Gráfico protótipo](#)

Métodos disponíveis:

- [graphprototype.create](#) - criando novos protótipos de gráficos
- [graphprototype.delete](#) - excluindo protótipos de gráficos
- [graphprototype.get](#) - recuperando protótipos de gráficos
- [graphprototype.update](#) - atualizando protótipos de gráficos

> Objeto protótipo gráfico

Os objetos a seguir estão diretamente relacionados à API graphprototype.

Protótipo de gráfico

O objeto de protótipo de gráfico tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----|-----| |graphid|string|(readonly) ID do protótipo do gráfico. | **altura**
(obrigatório)|inteiro|Altura do protótipo do gráfico em pixels. | **nome**
(obrigatório)|string|Nome do protótipo do gráfico. | **largura**
(obrigatório)|inteiro|Largura do protótipo do gráfico em pixels. | |graphtype|inteiro|Tipo de layout dos protótipos de gráfico.

Valores possíveis:
0 - (padrão) normal;
1 - empilhado;
2 - pizza;
3 - explodiu. | |percent_left|float|Percentil esquerdo.

Padrão: 0. | |percent_right|float|Percentil direito.

Padrão: 0. | |show_3d|inteiro|Se mostrar gráficos de pizza descobertos e explodidos em 3D.

Valores possíveis:
0 - (padrão) mostrar em 2D;
1 - mostrar em 3D. | |show_legend|integer|Se mostrar a legenda no gráfico descoberto.

Valores possíveis:
0 - ocultar;
1 - (padrão) show. | |show_work_period|integer|Se mostrar o tempo de trabalho no gráfico descoberto.

Valores possíveis:
0 - ocultar;
1 - (padrão) mostrar. | |templateid|string|(readonly) ID do protótipo do gráfico do template pai. | |yaxismax|float|O valor máximo fixo para o eixo Y. | |yaxismin|float|O valor mínimo fixo para o eixo Y. | |ymax_itemid|string|ID do item que é usado como o valor máximo para o eixo Y. | |ymax_type|inteiro|Método de cálculo do valor máximo para o eixo Y.

Valores possíveis:
0 - (padrão) calculado;
1 - fixo;
2 - item . | |ymin_itemid|string|ID do item que é usado como valor mínimo para o eixo Y. | |ymin_type|inteiro|Método de cálculo do valor mínimo para o eixo Y.

Valores possíveis:
0 - (padrão) calculado;
1 - fixo;
2 - item . | |discover|inteiro|Status de descoberta do protótipo do gráfico.

Valores possíveis:
0 - (padrão) novos gráficos serão descobertos;
1 - novos gráficos não serão descobertos e os gráficos existentes serão ser marcado como perdido. | |uuid|string|Identificador único universal, usado para vincular protótipos de gráficos importados aos já existentes. Usado apenas para protótipos de gráficos em modelos. Gerado automaticamente, se não fornecido.

Para operações de atualização, este campo é somente leitura. |

graphprototype.create

Descrição

object graphprototype.create(object/array graphPrototypes)

Este método permite criar novos protótipos de gráficos.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Protótipos de gráfico para criar.

Além do **protótipo de gráfico padrão properties**, o método aceita o seguinte parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----|-----| | **gitens**
(obrigatório)|array|Gráfico **itens** a ser criado para os protótipos do gráfico. Itens de gráfico podem referenciar itens e protótipos de itens, mas pelo menos um protótipo de item deve estar presente. |

Valores de retorno

(object) Retorna um objeto contendo os IDs do gráfico criado protótipos sob a propriedade graphids. A ordem dos IDs retornados corresponde à ordem dos protótipos de grafos passados.

Exemplos

Criando um protótipo de gráfico

Crie um protótipo de gráfico com dois itens.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "graphprototype.create",
  "parâmetros": {
    "name": "Uso de espaço em disco {#FSNAME}",
    "largura": 900,
    "altura": 200,
    "gitens": [
      {
        "itemid": "22828",
        "cor": "00AA00"
      },
    ],
  },
}
```

```

        {
            "itemid": "22829",
            "cor": "3333FF"
        }
    ],
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "grafos": [
            "652"
        ]
    },
    "id": 1
}

```

Veja também

- [Item de gráfico](#)

Fonte

CGraphPrototype::create() in ui/include/classes/api/services/CGraphPrototype.php.

graphprototype.delete

Descrição

object graphprototype.delete(array graphPrototypeIds)

Este método permite deletar protótipos de grafos.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos protótipos de gráfico a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs do gráfico excluído protótipos sob a propriedade graphids.

Exemplos

Excluindo vários protótipos de gráficos

Exclua dois protótipos de gráfico.

Solicitação:

```

{
    "jsonrpc": "2.0",
    "method": "graphprototype.delete",
    "parâmetros": [
        "652",
        "653"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Resposta:

```
{
    "jsonrpc": "2.0",
    "resultado": {
        "grafos": [
            "652",
            "653"
        ]
    },
    "id": 1
}
```

Fonte

CGraphPrototype::delete() in ui/include/classes/api/services/CGraphPrototype.php.

graphprototype.get

Descrição

integer/array graphprototype.get(object parameters)

O método permite recuperar protótipos de grafos de acordo com as parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [discoveryids|string/array|Retorna apenas protótipos de gráfico que pertencem às regras de descoberta fornecidas.] [graphids|string/array|Retorna apenas protótipos de gráfico com os IDs fornecidos.] [groupids|string/array|Retorna apenas protótipos de gráfico que pertencem a hosts nos grupos de hosts fornecidos.] [hostids|string/array|Retorna apenas protótipos de gráfico que pertencem aos hosts fornecidos.] [inherited|boolean|Se definido como true, retornará apenas protótipos de gráfico herdados de um modelo.] [itemids|string/array|Retorna apenas protótipos de gráfico que contenham os protótipos de itens fornecidos.] [templated|boolean|Se definido como true, retornará apenas protótipos de gráficos que pertencem a templates.] [templateids|string/array|Retorna apenas protótipos de gráfico que pertencem aos modelos fornecidos.] [selectDiscoveryRule|query|Retorne uma propriedade **discoveryRule** com a regra LLD à qual o protótipo do gráfico pertence.] [selectGraphItems|query|Retorne uma propriedade **gitems** com os itens do gráfico usados no protótipo do gráfico.] [selectGroups|query|Retorne uma propriedade **groups** com os grupos de hosts aos quais o protótipo gráfico pertence.] [selectHosts|query|Retorne uma propriedade **hosts** com os hosts aos quais o protótipo gráfico pertence.] [selectItems|query|Retorne uma propriedade **items** com os **items** e **item prototypes** usados no protótipo do gráfico.] [selectTemplates|query|Retorne uma propriedade **templates** com os templates aos quais o protótipo gráfico pertence.] [filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

 Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.

 Suporta filtros adicionais:
host - nome técnico do host ao qual o protótipo gráfico pertence;
hostid - ID do host ao qual o protótipo gráfico pertence.
sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

 Os valores possíveis são: graphid, name e graphtype.
countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no [comentário de referência](#).
editável|boolean|^|
excludePesquisa|boolean|^|
limite|inteiro|^|
saída|consulta|^|
preservekeys|boolean|^|
pesquisar|objeto|^|
searchByAny|boolean|^|
searchWildcardsEnabled|boolean|^|
ordem de classificação|cadeia/matriz|^|
startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando protótipos de gráfico de uma regra LLD

Recupere todos os protótipos de gráfico de uma regra LLD.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "graphprototype.get",
  "parâmetros": {
    "saída": "estender",
    "discoveryids": "27426"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "grafid": "1017",
      "parent_itemid": "27426",
      "name": "Uso de espaço em disco {#FSNAME}",
      "largura": "600",
      "altura": "340",
      "yaxismin": "0,0000",
      "yaxismax": "0,0000",
      "templateid": "442",
      "show_work_period": "0",
      "show_triggers": "0",
      "graphtype": "2",
      "show_legend": "1",
      "show_3d": "1",
      "percent_left": "0,0000",
      "percent_right": "0,0000",
      "ymin_type": "0",
      "ymax_type": "0",
      "ymin_itemid": "0",
      "ymax_itemid": "0",
      "descobrir": "0"
    }
  ],
  "id": 1
}
```

Veja também

- [Descoberta regra](#)
- [Item de gráfico](#)
- [Item](#)
- [Host](#)
- [Grupo de hosts](#)
- [Modelo](#)

Fonte

CGraphPrototype::get() in ui/include/classes/api/services/CGraphPrototype.php.

graphprototype.update

Descrição

object graphprototype.update(object/array graphPrototypes)

Este método permite atualizar protótipos de grafos existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do protótipo do gráfico a serem atualizadas.

A propriedade `graphid` deve ser definida para cada protótipo de gráfico, todas outras propriedades são opcionais. Somente as propriedades passadas serão atualizadas, todos os outros permanecerão inalterados.

Além do **protótipo de gráfico padrão properties**, o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [gitems|array] Gráfico **items** para substituir itens de gráfico existentes. Se um item de gráfico tiver a propriedade `gitemid` definida, ele será atualizado, caso contrário, um novo item de gráfico será criado.]

Valores de retorno

(object) Retorna um objeto contendo os IDs do gráfico atualizado protótipos sob a propriedade `graphids`.

Exemplos

Alterando o tamanho de um protótipo de gráfico

Altere o tamanho de um protótipo de gráfico para 1100 a 400 pixels.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "graphprototype.update",
  "parâmetros": {
    "grafid": "439",
    "largura": 1100,
    "altura": 400
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "grafos": [
      "439"
    ]
  },
  "id": 1
}
```

Fonte

`CGraphPrototype::update()` em `ui/include/classes/api/services/CGraphPrototype.php`.

Protótipo de host

Esta classe foi projetada para trabalhar com protótipos de host.

Referências de objetos:

- [Hospedeiro protótipo](#)
- [Protótipo do host inventário](#)
- [Link do grupo](#)
- [Grupo protótipo](#)

Métodos disponíveis:

- **hostprototype.create** - criando novos protótipos de host
- **hostprototype.delete** - excluindo protótipos de host
- **hostprototype.get** - recuperando protótipos de host
- **hostprototype.update** - atualizando protótipos de host

> Objeto protótipo do host

Os objetos a seguir estão diretamente relacionados à API **hostprototype**.

Protótipo de host

O objeto de protótipo do host tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |hostid|string|(readonly) ID do protótipo do host.| **host**
(obrigatório)|string|Nome técnico do protótipo do host.| |name|string|Nome visível do protótipo do host.

Padrão: valor da propriedade host.| |status|inteiro|Status do protótipo do host.

Os valores possíveis são:
0 - (padrão) host monitorado;
1 - host não monitorado.| |inventário_modo|inteiro|Modo de preenchimento do inventário do host.

Os valores possíveis são:
-1 - (padrão) desativado;
0 - manual;
1 - automático.| |templateid|string|(readonly) ID do protótipo do host do modelo pai.| |discover|integer|Status de descoberta do protótipo do host.

Valores possíveis:
0 - (padrão) novos hosts serão descobertos;
1 - novos hosts não serão descobertos e os hosts existentes serão ser marcado como perdido.| |custom_interfaces|integer|Fonte de interfaces para hosts criados pelo protótipo do host.

Valores possíveis:
0 - (padrão) herdam interfaces do host pai;
1 - use host protótipos de interfaces personalizadas.| |uuid|string|Identificador único universal, usado para vincular protótipos de host importados aos já existentes. Usado apenas para protótipos de host em modelos. Gerado automaticamente, se não fornecido.

Para operações de atualização, este campo é somente leitura.|

Link do grupo

O objeto de link de grupo vincula um protótipo de host a um grupo de hosts e tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |group_prototypeid|string|(readonly) ID do link do grupo.| **groupid**
(obrigatório)|string|ID do grupo de hosts.| |hostid|string|(readonly) ID do protótipo do host| |templateid|string|(readonly) ID do link do grupo de template pai.|

Protótipo de grupo

O objeto protótipo de grupo define um grupo que será criado para um host descoberto e tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |group_prototypeid|string|(readonly) ID do protótipo do grupo.| **nome**
(obrigatório)|string|Nome do protótipo do grupo.| |hostid|string|(readonly) ID do protótipo do host| |templateid|string|(readonly) ID do protótipo do grupo de template pai.|

Tag do protótipo do host

O objeto de tag de protótipo do host tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |tag
(obrigatório)|string|Nome da tag do protótipo do host.| |value|string|Valor da tag do protótipo do host.|

Interface personalizada

O objeto de interface personalizado tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |dns|string|nome DNS usado pela interface.

Obrigatório se a conexão for feita via DNS. Pode conter macros.| |ip|string|endereço IP usado pela interface.

Obrigatório se a conexão for feita via IP. Pode conter macros.| **main**
(obrigatório)|integer|Se a interface é usada como padrão no host. Apenas uma interface de algum tipo pode ser definida como padrão em um host.

Os valores possíveis são:
0 - não padrão;
1 - padrão.| **port**
(obrigatório)|string|Número da porta usada pela interface. Pode conter macros de usuário e LLD.| **tipo**
(obrigatório)|inteiro|Tipo de interface.

Os valores possíveis são:
1 - agente;
2 - SNMP;
3 - IPMI;
4 - JMX.
| **useip**
(obrigatório)|inteiro|Se a conexão deve ser feita via IP.

Os valores possíveis são:
0 - conectar usando o nome DNS do host;
1 - conecte-se usando o endereço IP do host para esta interface do host.| |details|array|Objeto adicional para interface. **Obrigatório** se o 'tipo' da interface for SNMP.|

Detalhes da interface personalizada

O objeto de detalhes tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |versão
(obrigatório)|inteiro|versão da interface SNMP.

Os valores possíveis são:
1 - SNMPv1;
2 - SNMPv2c;
3 - SNMPv3| |bulk|inteiro|Se usar

solicitações SNMP em massa.

Os valores possíveis são:
0 - não usar solicitações em massa;
1 - (padrão) - usar solicitações em massa.| |comunidade|string|comunidade SNMP. Usado apenas por interfaces SNMPv1 e SNMPv2.| |security-name|string|nome de segurança SNMPv3. Usado apenas por interfaces SNMPv3.| |nível de segurança|inteiro|nível de segurança SNMPv3. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - noAuthNoPriv;
1 - authNoPriv;
2 - authPriv. | |authpassphrase|string|Senha de autenticação SNMPv3. Usado apenas por interfaces SNMPv3.| |privpassphrase|string|Senha de privacidade SNMPv3. Usado apenas por interfaces SNMPv3.| |authprotocol|inteiro|protocolo de autenticação SNMPv3. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - MD5;
1 - SHA1;
2 - SHA224;
3 - SHA256;
4 - SHA384;
5 - SHA512.| |privprotocol|inteiro|SNMPv3 protocolo de privacidade. Usado apenas por interfaces SNMPv3.

Os valores possíveis são:
0 - (padrão) - DES;
1 - AES128;
2 - AES192;
3 - AES256;
4 - AES192C;
5 - AES256C.| |contextname|string|SNMPv3 nome do contexto. Usado apenas por interfaces SNMPv3.|

hostprototype.create

Descrição

`object hostprototype.create(object/array hostPrototypes)`

Este método permite criar novos protótipos de host.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Protótipos de host para criar.

Além do [protótipo de host padrão properties](#), o método aceita o seguinte parâmetros.

|Parâmetro|Tipo|Descrição| |-----|-----| |groupLinks
(obrigatório)|array|Grupo links a ser criado para o protótipo do host.| |ruleid
(obrigatório)|string|ID da regra LLD à qual o protótipo do host pertence.| |group-Prototypes|array|Grupo [protótipos](#) a ser criado para o protótipo do host.| |macros|object/array|[Macros do usuário](#) a serem criadas para o protótipo do host.| |tags|objeto/array|Protótipo de host [tags](#).| |interfaces|object/array|Protótipo de host [interfaces personalizadas](#).| |templates|object/array|[Templates](#) a serem vinculados ao protótipo do host.

Os templates devem ter a propriedade templateid definida.|

Valores de retorno

(object) Retorna um objeto contendo os IDs do host criado protótipos sob a propriedade `hostids`. A ordem dos IDs retornados corresponde à ordem dos protótipos de host passados.

Exemplos

Criando um protótipo de host

Crie um protótipo de host "{#VM.NAME}" na regra LLD "23542" com um grupo protótipo "{#HV.NAME}", par de tags "Datacenter": "{#DATACENTER.NAME}" e interface SNMPv2 personalizada 127.0.0.1:161 com comunidade {\$SNMP_COMMUNITY}. Vincule-o ao grupo de hosts "2".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "hostprototype.create",
  "parâmetros": {
    "host": "{#VM.NAME}",
    "ruleid": "23542",
    "custom_interfaces": "1",
    "groupLinks": [
      {
        "groupid": "2"
      }
    ],
    "grupoProtótipos": [
      {
        "name": "{#HV.NAME}"
      }
    ]
  }
}
```



```

    ],
    "Tag": [
        {
            "tag": "Datacenter",
            "valor": "#{#DATACENTER.NAME}"
        }
    ],
    "interfaces": [
        {
            "principal": "1",
            "tipo": "2",
            "useip": "1",
            "ip": "127.0.0.1",
            "dns": "",
            "porta": "161",
            "detalhes": {
                "versão": "2",
                "em massa": "1",
                "community": "{$SNMP_COMMUNITY}"
            }
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": {
        "hostids": [
            "10103"
        ]
    },
    "id": 1
}

```

Veja também

- [Link do grupo](#)
- [Protótipo do grupo](#)
- [Tag de protótipo do host](#)
- [Interface personalizada](#)
- [Do utilizador macro](#)

Fonte

CHostPrototype::create() em ui/include/classes/api/services/CHostPrototype.php.

hostprototype.delete

Descrição

object hostprototype.delete(array hostPrototypeIds)

Este método permite excluir protótipos de host.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos protótipos de host a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host excluído protótipos sob a propriedade `hostids`.

Exemplos

Excluindo vários protótipos de host

Exclua dois protótipos de host.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostprototype.delete",
  "parâmetros": [
    "10103",
    "10105"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10103",
      "10105"
    ]
  },
  "id": 1
}
```

Fonte

`CHostPrototype::delete()` in `ui/include/classes/api/services/CHostPrototype.php`.

hostprototype.get

Descrição

`integer/array hostprototype.get(object parameters)`

O método permite recuperar protótipos de host de acordo com o parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [hostids|string/array|Retorna apenas protótipos de host com os IDs fornecidos.] [discoveryids|string/array|Retorna apenas o protótipo do host que pertence às regras LLD fornecidas.] [inherited|boolean|Se definido como `true`, retornará apenas itens herdados de um modelo.] [selectDiscoveryRule|query|Retorne uma propriedade `discoveryRule` com a regra LLD à qual o protótipo do host pertence.] [selectInterfaces|query|Retorne uma propriedade `interfaces` com interfaces customizadas de protótipo de host.] [selectGroupLinks|query|Retorne uma propriedade `groupLinks` com os links de grupo do protótipo do host.] [selectGroupPrototypes|query|Retorne uma propriedade `groupPrototypes` com os protótipos de grupo do protótipo do host.] [selectMacros|query|Retorne uma propriedade `macros` com macros de protótipo de host.] [selectParentHost|query|Retorne uma propriedade `parentHost` com o host ao qual o protótipo do host pertence.] [selectTags|query|Retorne uma propriedade `tags` com tags de protótipo do host.] [selectTemplates|query|Retorne uma propriedade `templates` com os modelos vinculados ao protótipo do host.]

Suporta `count`.| [sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.]

Os valores possíveis são: `hostid`, `host`, `name` e `status`.| [countOutput|boolean|Estes

parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página [Informações genéricas da API Zabbix](#).
editável	boolean	^		excludePesquisa	boolean	^		filtro	objeto	^		limite	inteiro	^		saída	consulta	^		preservekeys	boolean	^
pesquisar	objeto	^		searchByAny	boolean	^		searchWildcardsEnabled	boolean	^		ordem de classificação	cadeia/matriz	^								
startSearch	boolean	^																				

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando protótipos de host de uma regra LLD

Recupere todos os protótipos de host, seus links de grupo, protótipos de grupo e tags de uma regra LLD.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "hostprototype.get",
  "parâmetros": {
    "saída": "estender",
    "selectInterfaces": "estender",
    "selectGroupLinks": "estender",
    "selectGroupPrototypes": "estender",
    "selectTags": "estender",
    "discoveryids": "23554"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "hostid": "10092",
      "host": "{#HV.UUID}",
      "name": "{#HV.UUID}",
      "estado": "0",
      "templateid": "0",
      "descobrir": "0",
      "custom_interfaces": "1",
      "inventory_mode": "-1",
      "groupLinks": [
        {
          "group_prototypeid": "4",
          "hostid": "10092",
          "groupid": "7",
          "templateid": "0"
        }
      ],
      "grupoProtótipos": [
        {
          "group_prototypeid": "7",
          "hostid": "10092",
          "name": "{#CLUSTER.NAME}",
          "templateid": "0"
        }
      ],
      "Tag": [
        {
          "tag": "Datacenter",

```

```

        "valor": "{#DATACENTER.NAME}"
    },
    {
        "tag": "Tipo de instância",
        "valor": "{#INSTANCE_TYPE}"
    }
],
"interfaces": [
    {
        "principal": "1",
        "tipo 2",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "porta": "161",
        "detalhes": {
            "versão 2",
            "em massa": "1",
            "community": "{$SNMP_COMMUNITY}"
        }
    }
]
}
],
"id": 1
}

```

Veja também

- [Link do grupo](#)
- [Protótipo do grupo](#)
- [Do utilizador macro](#)

Fonte

CHostPrototype::get() in ui/include/classes/api/services/CHostPrototype.php.

hostprototype.update

Descrição

object hostprototype.update(object/array hostPrototypes)

Este método permite atualizar protótipos de host existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades do protótipo do host a serem atualizadas.

A propriedade `hostid` deve ser definida para cada protótipo de host, todos os outros propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas os outros permanecerão inalterados.

Além do [protótipo de host padrão properties](#), o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| [groupLinks|array|Grupo [links](#) para substituir os links de grupo atuais no protótipo do host.] [groupPrototypes|array|Grupo [protótipos](#) para substituir os protótipos de grupo existentes no protótipo do host.] [macros|object/array|[Macros de usuário](#) para substituir as macros de usuário atuais.

Todas as macros que não estiverem listadas na solicitação serão removidas.] [tags|object/array|Protótipo de host [tags](#) para substituir as tags atuais.

Todas as tags que não estiverem listadas na solicitação serão removidas .] [interfaces|object/array|Protótipo de host [interfaces personalizadas](#) para substituir as interfaces atuais.

O objeto de interface personalizada deve conter todos os seus parâmetros.
>Todas as interfaces que não estão listadas na solicitação

serão removidas. | templates|object/array|**Templates** para substituir os templates atualmente vinculados.

Os templates devem ter a propriedade templateid definida.

Valores de retorno

(object) Retorna um objeto contendo os IDs do host atualizado protótipos sob a propriedade hostids.

Exemplos

Desativando um protótipo de host

Desabilite um protótipo de host, ou seja, defina seu status como 1.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "hostprototype.update",
  "parâmetros": {
    "hostid": "10092",
    "estado": 1
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10092"
    ]
  },
  "id": 1
}
```

Atualizando tags de protótipo de host

Substitua as tags de protótipo do host por novas.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "hostprototype.update",
  "parâmetros": {
    "hostid": "10092",
    "Tag": [
      {
        "tag": "Datacenter",
        "valor": "#{DATACENTER.NAME}"
      },
      {
        "tag": "Tipo de instância",
        "valor": "#{INSTANCE_TYPE}"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
```

```

        "10092"
    ]
},
"id": 1
}

```

Atualizando interfaces personalizadas de protótipo de host

Substitua interfaces herdadas por interfaces personalizadas de protótipo de host.

Solicitação:

```

{
  "jsonrpc": "2.0",
  "method": "hostprototype.update",
  "parâmetros": {
    "hostid": "10092",
    "custom_interfaces": "1",
    "interfaces": [
      {
        "principal": "1",
        "tipo": "2",
        "useip": "1",
        "ip": "127.0.0.1",
        "dns": "",
        "porta": "161",
        "detalhes": {
          "versão": "2",
          "em massa": "1",
          "community": "${SNMP_COMMUNITY}"
        }
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": {
    "hostids": [
      "10092"
    ]
  },
  "id": 1
}

```

Veja também

- [Link do grupo](#)
- [Protótipo do grupo](#)
- [Tag de protótipo do host](#)
- [Interface personalizada](#)
- [Do utilizador macro](#)

Fonte

CHostPrototype::update() em ui/include/classes/api/services/CHostPrototype.php.

Protótipo de item

Esta classe foi projetada para trabalhar com protótipos de itens.

Referências de objetos:

- **Item protótipo**

Métodos disponíveis:

- **itemprototype.create** - criando novos protótipos de itens
- **itemprototype.delete** - excluindo protótipos de itens
- **itemprototype.get** - recuperando protótipos de itens
- **itemprototype.update** - atualizando protótipos de itens

> Objeto protótipo de item

Os objetos a seguir estão diretamente relacionados à API `itemprototype`.

Protótipo de item

O objeto protótipo de item tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----|-----|-----| itemid|string|(readonly) ID do protótipo do item. | **atraso**
(obrigatório)|string|Intervalo de atualização do protótipo do item. Aceita segundos ou uma unidade de tempo com sufixo (30s,1m,2h,1d).
Opcionalmente, um ou mais **intervalos personalizados** podem ser especificados como intervalos flexíveis ou agendamento .
Vários intervalos são separados por um ponto e vírgula.
Macros de usuário e macros LLD podem ser usados. Uma única macro deve preencher todo o campo. Não há suporte para várias macros em um campo ou macros misturadas com texto.
Intervalos flexíveis podem ser escritos como duas macros separadas por uma barra (por exemplo, {\$FLEX_INTERVAL}/{FLEX_PERIOD}).

Opcional para Zabbix trapper, itens dependentes e para Zabbix agent (ativo) com chave `mqtt.get`. | **hostid**
(obrigatório)|string|ID do host ao qual o protótipo do item pertence.

Para operações de atualização, este campo é readonly. | **ruleid**
(obrigatório)|string|ID da regra LLD à qual o item pertence.

Para operações de atualização, este campo é readonly. | **interfaceid**
(obrigatório)|string|ID da interface do host do protótipo do item. Usado apenas para protótipos de itens de host.

Opcional para agente Zabbix (ativo), Zabbix interno, Zabbix trapper, item dependente, monitor de banco de dados e protótipos de itens calculados. | **chave**
(obrigatório)|string|Chave de protótipo do item. | **nome**
(obrigatório)|string|Nome do protótipo do item. | **tipo**
(obrigatório)|inteiro|Tipo do protótipo do item.

Valores possíveis:
0 - Zabbix agent;
2 - Zabbix trapper;
3 - verificação simples;
5 - Zabbix interno;
7 - agente Zabbix (ativo);
10 - verificação externa;
11 - monitor de banco de dados;
12 - agente IPMI;
13 - Agente SSH;
14 - Agente TELNET;
15 - calculado;
16 - Agente JMX;
17 - Trap SNMP;
18 - Item dependente;
19 - HTTP agente;
20 - Agente SNMP;
21 - Script. | **url**
(obrigatório)|string|string de URL necessária apenas para protótipos de item de agente HTTP. Suporta macros LLD, macros de usuário, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}. | **valor_tipo**
(obrigatório)|inteiro|Tipo de informação do protótipo do item.

Valores possíveis:
0 - número flutuante;
1 - caractere;
2 - log;
3 - numérico sem sinal;
4 - texto. | **allow_traps**|integer|campo de protótipo de item do agente HTTP. Permitir preencher valor como no tipo de item trapper também.

0 - (padrão) Não permitir aceitar dados de entrada.
1 - Permitir aceitar dados de entrada. | **auth_type**|integer|Usado apenas por protótipos de item de agente SSH ou protótipos de item de agente HTTP.

Método de autenticação de agente SSH valores possíveis:
0 - (padrão) senha;
1 - chave pública .

Método de autenticação do agente HTTP valores possíveis:
0 - (padrão) nenhum
1 - básico
2 - NTLM
3 - Kerberos | **description**|string|Descrição do protótipo do item. | **follow_redirects**|integer|campo de protótipo de item do agente HTTP. Siga os redirecionamentos de resposta ao agrupar dados.

0 - Não siga os redirecionamentos.
1 - (padrão) Siga os redirecionamentos. | **headers**|object|campo protótipo de item do agente HTTP. Objeto com cabeçalhos de solicitação HTTP(S), onde o nome do cabeçalho é usado como chave e o valor do cabeçalho como valor.

Exemplo:
{ "User-Agent": "Zabbix" } | **history**|string|Uma unidade de tempo de quanto tempo os dados do histórico devem ser armazenados. Também aceita macro de usuário e macro LLD.

Padrão: 90d. | **http_proxy**|string|Campo de protótipo de item do agente HTTP. Cadeia de conexão do proxy HTTP(S). | **ipmi_sensor**|string|sensor IPMI. Usado apenas por protótipos de itens do IPMI. | **jmx_endpoint**|string|String de conexão personalizada do agente JMX.

Valor padrão:
service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT }/jmxrmi | **logtimefmt**|string|Formato da hora nas entradas de log. Usado apenas por protótipos de itens de log. | **master_itemid**|integer|ID do item mestre.

Recurso de até 3 itens dependentes e protótipos de item e contagem máxima de itens dependentes e protótipos de item igual a 29999 são permitidos.

Requerido por itens dependentes. | **output_format**|integer|campo de protótipo de item do agente HTTP. A resposta deve ser convertida em JSON.

0 - (padrão) Armazenar raw.
1 - Converter em JSON. | **params**|string|Parâmetros adicionais dependendo do tipo de protótipo de item:
- script executado para protótipos de itens SSH e Telnet;
- consulta SQL para protótipos de itens de monitor de banco de dados;
- fórmula para protótipos de itens calculados . | **parameters**|array|Parâmetros adicionais para protótipos de itens de script. Matriz de objetos com propriedades 'nome' e 'valor', onde o nome deve ser exclusivo. | **senha**|string|Senha para autenticação. Usado por verificação simples, SSH, Telnet, monitor de banco de dados, protótipos de item de agente JMX e HTTP. | **post_type**|integer|campo de protótipo de item do agente HTTP. Tipo de corpo de dados de postagem armazenado na propriedade de postagens.

0 - (padrão)

Dados brutos.
2 - Dados JSON.
3 - Dados XML.| |posts|string|campo de protótipo de item do agente HTTP. Dados do corpo da solicitação HTTP(S). Usado com post_type.| |privatekey|string|Nome do arquivo de chave privada.| |publickey|string|Nome do arquivo de chave pública.| |query_fields|array|campo de protótipo de item do agente HTTP. Parâmetros de consulta. Matriz de objetos com pares 'chave': 'valor', onde valor pode ser uma string vazia.| |request_method|integer|campo de protótipo de item do agente HTTP. Tipo de método de solicitação.

0 - (padrão) GET
1 - POST
2 - PUT
3 - HEAD| |recuperar_modo|inteiro|campo protótipo de item do agente HTTP. Qual parte da resposta deve ser armazenada.

0 - (padrão) Corpo.
1 - Cabeçalhos.
2 - O corpo e os cabeçalhos serão armazenados.

Para request_method HEAD somente 1 é o valor permitido.| |snmp_oid|string|SNMP OID.| |ssl_cert_file|string|campo de protótipo de item do agente HTTP. Caminho do arquivo de chave SSL pública.| |ssl_key_file|string|campo de protótipo de item do agente HTTP. Caminho do arquivo de chave SSL privada.| |ssl_key_password|string|campo de protótipo de item do agente HTTP. Senha para arquivo de chave SSL.| |status|inteiro|Status do protótipo do item.

Valores possíveis:
0 - (padrão) protótipo do item ativado;
1 - protótipo do item desativado;
3 - item não suportado protótipo.| |status_codes|string|campo de protótipo de item do agente HTTP. Intervalos de códigos de status HTTP necessários separados por vírgulas. Também suporta macros de usuário ou macros LLD como parte da lista separada por vírgulas.

Exemplo: 200.200-{\$M},{\$M},200-400| |templateid|string|(somente leitura) ID do protótipo do item de modelo pai.| |timeout|string|Tempo limite da solicitação de sondagem de dados do item. Usado para protótipos de agente HTTP e item de script. Suporta macros de usuário e macros LLD.

padrão: 3s
valor máximo: 60s| |trapper_hosts|string|Hosts permitidos. Usado por protótipos de item de trapper ou protótipos de item HTTP.| |trends|string|Uma unidade de tempo de quanto tempo os dados de tendências devem ser armazenados. Também aceita macro de usuário e macro LLD.

Padrão: 365d.| |unidades|string|Unidades de valor.| |username|string|Username para autenticação. Usado por verificação simples, SSH, Telnet, monitor de banco de dados, protótipos de item de agente JMX e HTTP.

Requerido por protótipos de item SSH e Telnet.| |uuid|string|Identificador único universal, usado para vincular protótipos de itens importados aos já existentes. Usado apenas para protótipos de itens em modelos. Gerado automaticamente, se não fornecido.

Para operações de atualização, este campo é somente leitura.| |valuemapid|string|ID do mapa de valores associado.| |verify_host|integer|campo de protótipo de item do agente HTTP. Validar o nome do host na URL está no campo Nome Comum ou em um campo Nome Alternativo do Assunto do certificado do host.

0 - (padrão) Não validar.
1 - Validar.
| |verify_peer|integer|campo de protótipo de item do agente HTTP. Validar é o certificado do host autêntico.

0 - (padrão) Não validar.
1 - Validar.| |discover|inteiro|Status de descoberta do protótipo do item.

Valores possíveis:
0 - (padrão) novos itens serão descobertos;
1 - novos itens não serão descobertos e os itens existentes serão ser marcado como perdido.|

Tag de protótipo de item

O objeto de tag de protótipo de item tem as seguintes propriedades.

|Propriedade|**Tipo**|Descrição| |-----|-----|-----|-----| |**tag**
(obrigatório)|string|Nome da tag do protótipo do item.| |valor|string|Valor da tag de protótipo do item.|

Pré-processamento de protótipo de item

O objeto de pré-processamento de protótipo de item tem as seguintes propriedades.

|Propriedade|**Tipo**|Descrição| |-----|-----|-----|-----| |**tipo**
(obrigatório)|inteiro|O tipo de opção de pré-processamento.

Valores possíveis:
1 - Multiplicador personalizado;
2 - Corte direito;
3 - Trim esquerdo;
4 - Trim;
5 - Correspondência de expressão regular;
6 - Booleano para decimal;
7 - Octal para decimal;
8 - Hexadecimal para decimal;
9 - Alteração simples;
10 - Alteração por segundo;
11 - XML XPath;
12 - JSONPath;
13 - No intervalo;
14 - Corresponde à expressão regular;
15 - Não corresponde à expressão regular;
16 - Verifique se há erro no JSON;
17 - Verifique se há erro no XML;
18 - Verifique se há erro usando a expressão regular;
19 - Descarte inalterado;
20 - Descartar inalterado com pulsação;
21 - JavaScript;
22 - Padrão Prometheus;
23 - Prometheus para JSON;
24 - CSV para JSON;
25 - Substituir;
26 - Verificação não suportada;
27 - XML para JSON.| |**params**
(obrigatório)|string|Parâmetros adicionais usados pela opção de pré-processamento. Vários parâmetros são separados pelo caractere LF (\n).| |**error_handler**
(obrigatório)|inteiro|Tipo de ação usado em caso de falha na etapa de pré-processamento.

Valores possíveis:
0 - Mensagem de erro é definida pelo servidor Zabbix;
1 - Descartar valor;
2 - Definir valor personalizado;
3 - Definir mensagem de erro personalizada.| |**error_handler_params**
(obrigatório)|string|Parâmetros do manipulador de erros. Usado com error_handler.

Deve estar vazio, se error_handler for 0 ou 1.
Pode estar vazio se, error_handler for 2.
Não pode estar vazio, se error_handler é 3.|

Os seguintes parâmetros e manipuladores de erros são suportados para cada tipo de pré-processamento.

|Tipo de pré-processamento|Nome|Parâmetro 1|Parâmetro 2|Parâmetro 3|Manipuladores de erro suportados| |-----|-----|-----|-----|-----| |1|Multiplicador personalizado|número^{1,6}|<|<|0, 1, 2, 3| |2|Corte direito|lista de caracteres²|<|<|<| |3|Corte esquerdo|lista de caracteres²|<|<|<| |4|Cortar|lista de caracteres²|<|<|<| |5|Expressão regular|padrão³|saída²|<|<|<|0, 1, 2, 3| |6|Booleano para decimal|<|<|<|<|0, 1, 2, 3| |7|Octal para decimal|<|<|<|<|0, 1, 2, 3| |8|Hexadecimal para decimal|<|<|<|<|<|0, 1, 2, 3| |9|Mudança simples|<|<|<|<|<|0, 1, 2, 3| |10|Mudança por segundo|<|<|<|<|<|0, 1, 2, 3| |11|XML XPath|caminho⁴|<|<|<|<|<|0, 1, 2, 3| |12|JSONPath|caminho⁴|<|<|<|<|<|0, 1, 2, 3| |13|No intervalo|min^{1,6}|max^{1,6}|<|<|<|<|<|0, 1, 2, 3| |14|Corresponde à expressão regular|padrão³|<|<|<|<|<|0, 1, 2, 3| |15|Não corresponde à expressão regular|padrão³|<|<|<|<|<|0, 1, 2, 3| |16|Verifique se há erro em JSON|path⁴|<|<|<|<|<|0, 1, 2, 3| |17|Verificar erros em XML|path⁴|<|<|<|<|<|0, 1, 2, 3| |18|Verifique se há erro usando expressão reg-

ular|pattern³|output²|<|0, 1, 2, 3| |19|Descartar inalterado|<|<|<|<| |20|Descartar inalterado com pulsação|segundos^{5,6}|<|<|<|
 |21|JavaScript|script²|<|<|<| |22|Padrão do Prometheus|padrão^{6,7}|valor, rótulo, função|saída^{8,9}|0, 1, 2, 3| |23|Prometheus
 para JSON|pattern^{6,17}|<|<|0, 1, 2, 3| |24|CSV para JSON|caractere²|caractere²|0,1|0, 1, 2, 3| |25|Substituir|sequência de
 pesquisa²|substituição²|<|<| |26|Verificação não suportada|<|<|<|1, 2, 3| |27|XML para JSON|<|<|<|0, 1, 2, 3|

¹ número inteiro ou de ponto flutuante

² sequência de caracteres

³ expressão regular

⁴ JSONPath ou XML XPath

⁵ inteiro positivo (com suporte de sufixos de tempo, por exemplo, 30s, 1m, 2h, 1d)

⁶ macro do usuário, macro LLD

⁷ Padrão do Prometheus seguindo a sintaxe: <nome da métrica>{<nome do rótulo>=<valor do rótulo>," ...} == <valor>. Cada Componente padrão do Prometheus (métrica, nome do rótulo, valor do rótulo e métrica value) pode ser macro de usuário ou macro LLD.

⁸ Saída do Prometheus seguindo a sintaxe: <label name> (pode ser uma macro de usuário ou uma macro LLD) se label for selecionado como o segundo parâmetro.

⁹ Uma das funções de agregação: sum, min, max, avg, count se function for selecionado como o segundo parâmetro.

itemprototype.create

Descrição

object itemprototype.create(object/array itemPrototypes)

Este método permite criar novos protótipos de itens.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Protótipo de item a ser criado.

Além do **protótipo de item padrão properties**, o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição| |-----|-----|-----|-----| |ruleid
(obrigatório)|string|ID da regra LLD à qual o item pertence.| |preprocessing|array|Protótipo de item **pré-processamento** opções.| |tags|array|Protótipo de item **tags**.|

Valores de retorno

(object) Retorna um objeto contendo os IDs do item criado protótipos sob a propriedade itemids. A ordem dos IDs retornados corresponde à ordem dos protótipos de itens passados.

Exemplos

Criando um protótipo de item

Crie um protótipo de item para monitorar o espaço livre em disco em um arquivo descoberto sistema. Os itens descobertos devem ser itens numéricos do agente Zabbix atualizados a cada 30 segundos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.create",
  "parâmetros": {
    "name": "Espaço livre em disco em {#FSNAME}",
    "key_": "vfs.fs.size[{#FSNAME},grátis]",
    "hostid": "10197",
    "ruleid": "27665",
    "tipo": 0,
    "value_type": 3,
    "interfaceid": "112",
    "atraso": "30s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
}
```

```
    "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "27666"
    ]
  },
  "id": 1
}
```

Criando um protótipo de item com pré-processamento

Crie um item usando a mudança por segundo e um multiplicador personalizado como segundo passo.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.create",
  "parâmetros": {
    "name": "Tráfego de rede de entrada em {#IFNAME}",
    "key_": "net.if.in[{#IFNAME}]",
    "hostid": "10001",
    "ruleid": "27665",
    "tipo": 0,
    "value_type": 3,
    "atraso": "60s",
    "unidades": "bps",
    "interfaceid": "1155",
    "pré-processando": [
      {
        "tipo": "10",
        "parâmetros": "",
        "error_handler": "0",
        "error_handler_params": ""
      },
      {
        "tipo": "1",
        "parâmetros": "8",
        "error_handler": "2",
        "error_handler_params": "10"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "44211"
    ]
  },
  "id": 1
}
```

Criando protótipo de item dependente

Crie um protótipo de item dependente para o protótipo de item mestre com ID 44211. Apenas dependências no mesmo host (regra de modelo/descoberta) são permitidas, portanto, o item mestre e dependente deve ter o mesmo hostid e ruleid.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.create",
  "parâmetros": {
    "hostid": "10001",
    "ruleid": "27665",
    "name": "Protótipo de item de teste dependente",
    "key_": "dependent.prototype",
    "tipo": "18",
    "master_itemid": "44211",
    "value_type": "3"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "44212"
    ]
  },
  "id": 1
}
```

Criar protótipo de item do agente HTTP

Crie protótipo de item com URL usando macro de usuário, campos de consulta e cabeçalhos.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.create",
  "parâmetros": {
    "tipo": "19",
    "hostid": "10254",
    "ruleid": "28256",
    "interfaceid": "2",
    "name": "exemplo de protótipo de item de API",
    "key_": "api_http_item",
    "value_type": "3",
    "url": "${URL_PROTOTYPE}",
    "campos_consulta": [
      {
        "min": "10"
      },
      {
        "máximo": "100"
      }
    ],
    "cabeçalhos": {
      "X-Source": "api"
    },
    "atraso": "35"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "28305"
    ]
  },
  "id": 1
}
```

Criar protótipo de item de script

Crie uma coleta de dados simples usando um protótipo de item de script.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.create",
  "parâmetros": {
    "name": "Exemplo de script",
    "key_": "custom.script.itemprototype",
    "hostid": "12345",
    "tipo": 21,
    "value_type": 4,
    "params": "var request = new CurlHttpRequest();\nreturn request.Post(\"https://postman-echo.com/post\");",
    "parâmetros": [{
      "nome": "host",
      "valor": "{HOST.CONN}"
    }],
    "tempo limite": "6s",
    "atraso": "30s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 2
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "items": [
      "23865"
    ]
  },
  "id": 3
}
```

Fonte

CIItemPrototype::create() in ui/include/classes/api/services/CIItemPrototype.php.

itemprototype.delete

Descrição

object itemprototype.delete(array itemPrototypeIds)

Este método permite excluir protótipos de itens.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(array) IDs dos protótipos de itens a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs do item excluído protótipos sob a propriedade `prototypeids`.

Exemplos

Excluindo vários protótipos de itens

Exclua dois protótipos de itens.

Os protótipos de itens dependentes são removidos automaticamente se o item mestre ou protótipo do item é excluído.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "itemprototype.delete",
  "parâmetros": [
    "27352",
    "27356"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "prototypeids": [
      "27352",
      "27356"
    ]
  },
  "id": 1
}
```

Fonte

`CItemPrototype::delete()` in `ui/include/classes/api/services/CItemPrototype.php`.

itemprototype.get

Descrição

`integer/array itemprototype.get(object parameters)`

O método permite recuperar protótipos de itens de acordo com o parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem o output desejado.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
<code>discoveryids</code>	<code>string/array</code>	Retorna apenas protótipos de item que pertencem às regras LLD fornecidas.
<code>graphids</code>	<code>string/array</code>	Retorna apenas protótipos de item que são usados nos protótipos de gráfico fornecidos.

Parâmetro	Tipo	Descrição
hostids	string/array	Retorna apenas protótipos de item que pertencem aos hosts fornecidos.
inherited	boolean	Se definido como <code>true</code> , retorna apenas protótipos de item herdados de um template.
itemids	string/array	Retorna apenas protótipos de item com os IDs fornecidos.
monitored	boolean	Se definido como <code>true</code> , retorna apenas protótipos de item habilitados que pertencem a hosts monitorados.
templated	boolean	Se definido como <code>true</code> , retorna apenas protótipos de item que pertencem a templates.
templateids	string/array	Retorna apenas protótipos de item que pertencem aos templates fornecidos.
triggerids	string/array	Retorna apenas protótipos de item que são usados nos protótipos de trigger fornecidos.
selectDiscoveryRule	query	Retorna uma propriedade <code>discoveryRule</code> com a regra de descoberta de baixo nível à qual o protótipo de item pertence.
selectGraphs	query	Retorna uma propriedade <code>graphs</code> com protótipos de gráfico que o protótipo de item usa.
selectHosts	query	Suporta <code>count</code> . Retorna uma propriedade <code>hosts</code> com um array de hosts aos quais o protótipo de item pertence.
selectTags	query	Retorna as tags do protótipo de item na propriedade <code>tags</code> .
selectTriggers	query	Retorna uma propriedade <code>triggers</code> com protótipos de trigger que o protótipo de item pertence.
		Suporta <code>count</code> .

Parâmetro	Tipo	Descrição
selectPreprocessing	query	<p>Retorna uma propriedade preprocessing com opções de pré-processamento de item.</p> <p>Possui as seguintes propriedades:</p> <p>type - (string) O tipo de opção de pré-processamento:</p> <ul style="list-style-type: none"> 1 - Custom multiplier; 2 - Right trim; 3 - Left trim; 4 - Trim; 5 - Regular expression matching; 6 - Boolean to decimal; 7 - Octal to decimal; 8 - Hexadecimal to decimal; 9 - Simple change; 10 - Change per second; 11 - XML XPath; 12 - JSONPath; 13 - In range; 14 - Matches regular expression; 15 - Does not match regular expression; 16 - Check for error in JSON; 17 - Check for error in XML; 18 - Check for error using regular expression; 19 - Discard unchanged; 20 - Discard unchanged with heartbeat; 21 - JavaScript; 22 - Prometheus pattern; 23 - Prometheus to JSON; 24 - CSV to JSON; 25 - Replace; 26 - Check for not supported value; 27- XML to JSON. <p>params - (string) Parâmetros adicionais usados pela opção de pré-processamento. Múltiplos parâmetros são separados por caractere LF (\n).</p> <p>error_handler - (string) Tipo de ação usada em caso de falha na etapa de pré-processamento:</p> <ul style="list-style-type: none"> 0 - Mensagem de erro definida pelo servidor Zabbix; 1 - Descartar valor; 2 - Definir valor personalizado; 3 - Definir mensagem de erro personalizada. <p>error_handler_params - (string) Parâmetros do manipulador de erros.</p>
selectValueMap	query	Retorna uma propriedade valuemap com o mapa de valor do protótipo de item.
filter	object	<p>Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.</p> <p>Aceita um array, onde as chaves são nomes de propriedades e os valores são um único valor ou um array de valores para correspondência.</p> <p>Suporta filtros adicionais:</p> <p>host - nome técnico do host ao qual o protótipo de item pertence.</p>
limitSelects	integer	<p>Limita o número de registros retornados por sub-seleções.</p> <p>Aplica-se às seguintes sub-seleções:</p> <p>selectGraphs - resultados serão ordenados por name;</p> <p>selectTriggers - resultados serão ordenados por description.</p>

Parâmetro	Tipo	Descrição
sortfield	string/array	Ordena o resultado pelas propriedades fornecidas.
countOutput	boolean	Valores possíveis são: itemid, name, key_, delay, type e status. Esses parâmetros sendo comuns a todos os métodos get são descritos em detalhes no reference commentary .
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- um array de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando protótipos de itens de uma regra LLD

Recupere todos os protótipos de itens para um ID de regra LLD específico

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.get",
  "params": {
    "output": "extend",
    "discoveryids": "27426"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "23077",
      "type": "0",
      "snmp_oid": "",
      "hostid": "10079",
      "name": "Incoming network traffic on en0",
      "key_": "net.if.in[en0]",
      "delay": "1m",
      "history": "1w",
      "trends": "365d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "bps",
      "logtimefmt": "",
      "templateid": "0",
      "valuemapid": "0",
      "params": ""
    }
  ]
}
```



```

    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "interfaceid": "0",
    "description": "",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "timeout": "3s",
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "discover": "0",
    "uuid": "",
    "parameters": []
},
{
    "itemid": "10010",
    "type": "0",
    "snmp_oid": "",
    "hostid": "10001",
    "name": "Processor load (1 min average per core)",
    "key_": "system.cpu.load[percpu,avg1]",
    "delay": "1m",
    "history": "1w",
    "trends": "365d",
    "status": "0",
    "value_type": "0",
    "trapper_hosts": "",
    "units": "",
    "logtimefmt": "",
    "templateid": "0",
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "interfaceid": "0",
    "description": "The processor load is calculated as system CPU load divided by number of CPU c",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",

```

```

        "timeout": "3s",
        "url": "",
        "query_fields": [],
        "posts": "",
        "status_codes": "200",
        "follow_redirects": "1",
        "post_type": "0",
        "http_proxy": "",
        "headers": [],
        "retrieve_mode": "0",
        "request_method": "0",
        "output_format": "0",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "lastclock": "0",
        "lastns": "0",
        "lastvalue": "0",
        "prevvalue": "0",
        "discover": "0",
        "uuid": "",
        "parameters": []
    }
],
    "id": 1
}

```

Encontrando item dependente

Encontre um item Dependente para o item com ID "25545".

Solicitação:

```

{
    "jsonrpc": "2.0",
    "método": "item.get",
    "parâmetros": {
        "saída": "estender",
        "filtro": {
            "tipo": "18",
            "master_itemid": "25545"
        },
        "limite": "1"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Resposta:

```

{
    "jsonrpc": "2.0",
    "resultado": [
        {
            "itemid": "25547",
            "tipo": "18",
            "snmp_oid": "",
            "hostid": "10116",
            "nome": "Segundos",
            "key_": "apache.status.uptime.seconds",
            "atraso": "0",
            "história": "90d",

```

```

        "tendências": "365d",
        "estado": "0",
        "value_type": "3",
        "trapper_hosts": "",
        "unidades": "",
        "Fórmula": "",
        "erro": "",
        "logtimefmt": "",
        "templateid": "0",
        "valuemap": "0",
        "parâmetros": "",
        "ipmi_sensor": "",
        "authtype": "0",
        "nome do usuário": "",
        "senha": "",
        "chave pública": "",
        "chave privada": "",
        "bandeiras": "0",
        "interfaceid": "0",
        "Descrição": "",
        "link_inventário": "0",
        "vida": "30d",
        "estado": "0",
        "evaltype": "0",
        "master_itemid": "25545",
        "jmx_endpoint": "",
        "master_itemid": "0",
        "tempo limite": "3s",
        "url": "",
        "query_fields": [],
        "Postagens": "",
        "status_codes": "200",
        "follow_redirects": "1",
        "post_type": "0",
        "proxy HTTP": "",
        "cabçalhos": [],
        "recuperar_modo": "0",
        "request_method": "0",
        "output_format": "0",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "último relógio": "0",
        "últimos": "0",
        "último valor": "0",
        "valor anterior": "0",
        "descobrir": "0",
        "parâmetros": []
    }
],
    "id": 1
}

```

Localizar protótipo de item do agente HTTP

Encontre o protótipo do item do agente HTTP com o método de solicitação HEAD para ID do host.

Solicitação:

```

{
    "jsonrpc": "2.0",

```

```

    "método": "itemprototype.get",
    "parâmetros": {
      "hostids": "10254",
      "filtro": {
        "tipo": "19",
        "request_method": "3"
      }
    },
    "id": 17,
    "auth": "d678e0b85688ce578ff061bd29a20d3b"
  }
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "itemid": "28257",
      "tipo": "19",
      "snmp_oid": "",
      "hostid": "10254",
      "nome": "descoberto",
      "key_": "item[{-#INAME}]",
      "atraso": "{#IUPDATE}",
      "história": "90d",
      "tendências": "30d",
      "estado": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "unidades": "",
      "Fórmula": "",
      "erro": "",
      "logtimefmt": "",
      "templateid": "28255",
      "valuemap": "0",
      "parâmetros": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "nome do usuário": "",
      "senha": "",
      "chave pública": "",
      "chave privada": "",
      "bandeiras": "2",
      "interfaceid": "2",
      "Descrição": "",
      "link_inventário": "0",
      "vida": "30d",
      "estado": "0",
      "evaltype": "0",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "tempo limite": "3s",
      "url": "{#IURL}",
      "query_fields": [],
      "Postagens": "",
      "status_codes": "",
      "follow_redirects": "0",
      "post_type": "0",
      "proxy HTTP": "",
      "cabecalhos": [],
      "recuperar_modulo": "0",
      "request_method": "3",
    }
  ]
}

```

```

        "output_format": "0",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "descobrir": "0",
        "parâmetros": []
    }
],
    "id": 17
}

```

Consulte também

- [Host](#)
- [Graph prototype](#)
- [Trigger prototype](#)

Fonte

CItemPrototype::get() in ui/include/classes/api/services/CItemPrototype.php.

itemprototype.update

Descrição

object itemprototype.update(object/array itemPrototypes)

Este método permite atualizar protótipos de itens existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. See [User roles](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do protótipo do item a serem atualizadas.

A propriedade `itemid` deve ser definida para cada protótipo de item, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de protótipo de item padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
preprocessing	array	Protótipo de item preprocessing opções para substituir as opções de pré-processamento atuais.
tags	array	Protótipo de item tags .

Valores de retorno

(object) Retorna um objeto contendo os IDs dos protótipos de itens atualizados na propriedade `itemids`.

Exemplos

Alterando a interface de um protótipo de item

Altere a interface do host que será usada pelos itens descobertos.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "itemprototype.update",
    "params": {
        "itemid": "27428",

```

```
    "interfaceid": "132"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "27428"
    ]
  },
  "id": 1
}
```

Atualizar protótipo de item dependente

Atualize o protótipo de item dependente com o novo ID de protótipo de item mestre. Apenas dependências no mesmo host (regra de modelo/descoberta) são permitidas, portanto, o item mestre e dependente deve ter o mesmo hostid e ruleid.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.update",
  "params": {
    "master_itemid": "25570",
    "itemid": "189030"
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "189030"
    ]
  },
  "id": 1
}
```

Atualizar protótipo de item do agente HTTP

Altere os campos de consulta e remova todos os cabeçalhos personalizados.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.update",
  "params": {
    "itemid": "28305",
    "query_fields": [
      {
        "random": "qwertyuiopasdfghjklzxcvbnm"
      }
    ],
    "headers": []
  },
  "auth": "700ca65537074ec963db7efabda78259",
}
```

```
    "id": 1
  }
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "28305"
    ]
  },
  "id": 1
}
```

Atualizando opções de pré-processamento de itens

Atualizar um protótipo de item com a regra de pré-processamento de item “Custom multiplier”.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.update",
  "params": {
    "itemid": "44211",
    "preprocessing": [
      {
        "type": 1,
        "params": "4",
        "error_handler": 2,
        "error_handler_params": "5"
      }
    ]
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "44211"
    ]
  },
  "id": 1
}
```

Atualizando um protótipo de item de script

Atualize um protótipo de item de script com um script diferente e remova parâmetros desnecessários que foram usados pelo script anterior.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.update",
  "params": {
    "itemid": "23865",
    "parameters": [],
    "script": "Zabbix.Log(3, 'Log test');\nreturn 1;"
  },
  "auth": "700ca65537074ec963db7efabda78259",
}
```

```
}
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "23865"
    ]
  },
  "id": 1
}
```

Fonte

CItemPrototype::update() in ui/include/classes/api/services/CItemPrototype.php.

Protótipo de trigger

Esta classe foi projetada para trabalhar com protótipos de trigger.

Referências de objetos:

- [Trigger prototype](#)

Métodos disponíveis:

- [triggerprototype.create](#) - criando novos protótipos de trigger
- [triggerprototype.delete](#) - excluindo protótipos de trigger
- [triggerprototype.get](#) - recuperando protótipos de trigger
- [triggerprototype.update](#) - atualizando protótipos de trigger

> Objeto de protótipo de trigger

Os objetos a seguir estão diretamente relacionados à API `triggerprototype`.

Protótipo de trigger

O objeto protótipo de trigger tem as seguintes propriedades.

Propriedade	Tipo	Descrição
triggerid	string	(somente leitura) ID do protótipo de trigger.
description (requerido)	string	Nome do protótipo da trigger.
expression (requerido)	string	Expressão de trigger reduzida.
event_name	string	Nome do evento gerado pela trigger.
opdata	string	Dados operacionais.
comments	string	Comentários adicionais ao protótipo de trigger.
priority	integer	Gravidade do protótipo de trigger. Valores possíveis: 0 - (padrão) não classificado; 1 - informação; 2 - aviso; 3 - média;< br>4 - alto; 5 - desastre.
status	integer	Se o protótipo de trigger está ativado ou desativado. Valores possíveis: 0 - (padrão) ativado; 1 - desativado.

Propriedade	Tipo	Descrição
templateid	string	(somente leitura) ID do protótipo de trigger do template pai.
templateid	string	(somente leitura) ID do protótipo de trigger do modelo pai.
type	integer	Se o protótipo de trigger pode gerar vários eventos de problema. Valores possíveis: 0 - (padrão) não gera vários eventos; 1 - gera vários eventos.
url	string	URL associado ao protótipo de trigger.
recovery_mode	integer	Modo de geração de eventos OK. Os valores possíveis são: 0 - (padrão) Expressão; 1 - Expressão de recuperação; 2 - Nenhum.
recovery_expression	string	Expressão de recuperação de trigger reduzida.
correlation_mode	integer	O evento OK fecha. Os valores possíveis são: 0 - (padrão) Todos os problemas; 1 - Todos os problemas se os valores das etiquetas corresponderem.
correlation_tag	string	Etiqueta para correspondência.
manual_close	integer	Permitir fechamento manual. Os valores possíveis são: 0 - (padrão) Não; 1 - Sim.
discover	integer	Status de descoberta do protótipo de trigger. Valores possíveis: 0 - (padrão) novas triggers serão descobertas; 1 - novas triggers não serão descobertas e as triggers existentes serão marcadas como perdidas.
uuid	string	Identificador único universal, usado para vincular protótipos de triggers importados aos já existentes. Usado apenas para protótipos de triggers em modelos. Gerado automaticamente, se não fornecido.
Para operações de atualização, este campo é somente leitura.		

Acionar etiqueta de protótipo

O objeto de etiqueta de protótipo de acionador tem as seguintes propriedades.

Propriedade	Type	Descrição
tag (requerido)	string	Acionar o nome da etiqueta do protótipo.
value	string	Valor de etiqueta de protótipo de trigger.

triggerprototype.create

Descrição

`object triggerprototype.create(object/array triggerPrototypes)`

Este método permite criar novos protótipos de trigger.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parameters

(object/array) Criar protótipos de trigger.

Além das **propriedades de protótipo de trigger padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
dependencies	array	Trigger e protótipos de trigger dos quais o protótipo de trigger depende.
tags	array	As triggers devem ter a propriedade triggerid definida. Etiquetas de Protótipo de trigger.

Attention:
A expressão de trigger deve ser fornecida em sua forma expandida e deve conter pelo menos um protótipo de item.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos protótipos de trigger criados na propriedade triggerids. A ordem dos IDs retornados corresponde à ordem dos protótipos de triggers passados.

Exemplos

Criando um protótipo de trigger

Crie um protótipo de trigger para detectar quando um sistema de arquivos tem menos de 20% de espaço livre em disco.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "triggerprototype.create",
  "params": {
    "description": "Free disk space is less than 20% on volume {#FSNAME}",
    "expression": "last(/Zabbix server/vfs.fs.size[{#FSNAME},pfree])<20",
    "tags": [
      {
        "tag": "volume",
        "value": "{#FSNAME}"
      },
      {
        "tag": "type",
        "value": "{#FSTYPE}"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "17372"
    ]
  },
  "id": 1
}
```

Fonte

CTriggerPrototype::create() in ui/include/classes/api/services/CTriggerPrototype.php.

triggerprototype.delete

Descrição

object triggerprototype.delete(array triggerPrototypeIds)

Este método permite deletar protótipos de trigger.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos protótipos de trigger a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos protótipos de trigger excluídos na propriedade triggerids.

Exemplos

Excluindo vários protótipos de gatilho

Excluir dois protótipos de gatilho.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "triggerprototype.delete",
  "params": [
    "12002",
    "12003"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "12002",
      "12003"
    ]
  },
  "id": 1
}
```

Fonte

CTriggerPrototype::delete() in ui/include/classes/api/services/CTriggerPrototype.php.

triggerprototype.get

Descrição

integer/array triggerprototype.get(object parameters)

O método permite recuperar protótipos de trigger de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Type	Descrição
active	flag	Retorna apenas protótipos de trigger habilitados que pertencem a hosts monitorados.
discoveryids	string/array	Retorna apenas protótipos de trigger que pertencem às regras LLD fornecidas.
functions	string/array	Retorne apenas triggers que usam as funções fornecidas.
group	string	Consulte a página supported trigger functions para obter uma lista de funções compatíveis. Retorna apenas protótipos de trigger que pertencem a hosts dos grupos de hosts com o nome fornecido.
groupids	string/array	Retorna apenas protótipos de trigger que pertencem a hosts dos grupos de hosts fornecidos.
host	string	Retorna apenas protótipos de trigger que pertencem a hosts com o nome fornecido.
inherited	boolean	Se definido como <code>true</code> , retorna apenas protótipos de trigger herdados de um modelo.
manutenção	boolean	Se definido como <code>true</code> retorna apenas protótipos de trigger habilitados que pertencem a hosts em manutenção.
min_severity	integer	Retorna apenas protótipos de trigger com gravidade maior ou igual à gravidade especificada.
monitored	flag	Retorna apenas protótipos de trigger ativados que pertencem a hosts monitorados e contêm apenas itens ativados.
templated	boolean	Se definido como <code>true</code> , retornará apenas protótipos de trigger que pertencem a templates.
templateids	string/array	Retorna apenas protótipos de trigger que pertencem aos modelos fornecidos.
triggerids	string/array	Retorna apenas protótipos de trigger com os IDs fornecidos.
expandExpression	flag	Expandir funções e macros na expressão da trigger.
selectDependencies	query	Retorne protótipos de trigger e triggers dos quais o protótipo de trigger depende na propriedade <code>dependencies</code> .
selectDiscoveryRule	query	Retorne a LLD rule à qual o protótipo da trigger pertence.
selectFunctions	query	Retorna as funções usadas no protótipo do trigger na propriedade <code>functions</code> .
Os objetos de função representam as funções usadas na expressão do trigger e possuem as seguintes propriedades:		
<code>functionid</code> - *(string)* ID da função;		
<code>itemid</code> - (string) ID do item usado na função;		
<code>function</code> - (string) nome da função;		
<code>parameter</code> - (string) parâmetro passado para a função. O parâmetro de consulta é substituído pelo símbolo <code>\$</code> na string retornada.		
selectGroups	query	Retorne os grupos de hosts aos quais o protótipo do trigger pertence na propriedade groups .
selectHosts	query	Retorne os hosts aos quais o protótipo do trigger pertence na propriedade hosts .
selectItems	query	Itens de retorno e protótipos de itens usaram o protótipo de trigger na propriedade items .
selectTags	query	Retorne as etiquetas de protótipo de trigger na propriedade etiquetas .
filter	object	Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.
Aceita uma array, em que as chaves são nomes de propriedades e os valores são um único valor ou uma array de valores para correspondência.		
Suporta filtros adicionais:		
<code>host</code> - nome técnico do host ao qual o protótipo da trigger pertence;		
<code>hostid</code> - ID do host ao qual o protótipo da trigger pertence.		

Parâmetro	Type	Descrição
limitSelects	integer	Limite o número de registros retornados por subseleções.
sortfield	string/array	<p>Aplica-se às seguintes subseleções:</p> <p><code>selectHosts</code> - os resultados serão classificados por <code>host</code>.</p> <p>Ordenar o resultado pelas propriedades fornecidas.</p> <p>Os valores possíveis são: <code>triggerid</code>, <code>description</code>, <code>status</code> e <code>priority</code>.</p>
countOutput	boolean	Esses parâmetros sendo comuns para todos os métodos <code>get</code> são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro `countOutput` foi usado.

Exemplos

Recuperar protótipos de trigger de uma regra LLD

Recupere todos os protótipos de trigger e suas funções de uma regra LLD.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "triggerprototype.get",
  "params": {
    "output": "extend",
    "selectFunctions": "extend",
    "discoveryids": "22450"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "triggerid": "13272",
      "expression": "{12598}<20",
      "description": "Free inodes is less than 20% on volume {#FSNAME}",
      "url": "",
      "status": "0",
      "priority": "2",
      "comments": "",
      "templateid": "0",
      "type": "0",
      "flags": "2",

```

```

        "recovery_mode": "0",
        "recovery_expression": "",
        "correlation_mode": "0",
        "correlation_tag": "",
        "manual_close": "0",
        "opdata": "",
        "discover": "0",
        "functions": [
            {
                "functionid": "12598",
                "itemid": "22454",
                "triggerid": "13272",
                "parameter": "$",
                "function": "last"
            }
        ]
    },
    {
        "triggerid": "13266",
        "expression": "{13500}<20",
        "description": "Free disk space is less than 20% on volume {#FSNAME}",
        "url": "",
        "status": "0",
        "priority": "2",
        "comments": "",
        "templateid": "0",
        "type": "0",
        "flags": "2",
        "recovery_mode": "0",
        "recovery_expression": "",
        "correlation_mode": "0",
        "correlation_tag": "",
        "manual_close": "0",
        "opdata": "",
        "discover": "0",
        "functions": [
            {
                "functionid": "13500",
                "itemid": "22686",
                "triggerid": "13266",
                "parameter": "$",
                "function": "last"
            }
        ]
    }
],
    "id": 1
}

```

Recuperando um protótipo de trigger específica com etiquetas

Request:

```

{
    "jsonrpc": "2.0",
    "method": "triggerprototype.get",
    "params": {
        "output": [
            "triggerid",
            "description"
        ]
        "selectTags": "extend",
        "triggerids": [
            "17373"
        ]
    }
}

```

```

    ],
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "triggerid": "17373",
      "description": "Free disk space is less than 20% on volume {#FSNAME}",
      "tags": [
        {
          "tag": "volume",
          "value": "{#FSNAME}"
        },
        {
          "tag": "type",
          "value": "{#FSTYPE}"
        }
      ]
    }
  ],
  "id": 1
}

```

Veja também

- [Discovery rule](#)
- [Item](#)
- [Host](#)
- [Host group](#)

Fonte

CTriggerPrototype::get() in ui/include/classes/api/services/CTriggerPrototype.php.

triggerprototype.update

Descrição

object triggerprototype.update(object/array triggerPrototypes)

Este método permite atualizar os protótipos de trigger existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parameters

(object/array) [Trigger prototype properties](#) to be updated.

The triggerid property must be defined for each trigger prototype, all other properties are optional. Only the passed properties will be updated, all others will remain unchanged.

Additionally to the [standard trigger prototype properties](#) the method accepts the following parameters.

Parameter	Type	Description
dependencies	array	Triggers and trigger prototypes that the trigger prototype is dependent on.
tags	array	The triggers must have the triggerid property defined. Trigger prototype tags .

Attention:

The trigger expression has to be given in its expanded form and must contain at least one item prototype.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos protótipos de triggers atualizados na propriedade `triggerids`.

Exemplos

Habilitando um protótipo de trigger

Habilite um protótipo de trigger, ou seja, defina seu status para 0.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "triggerprototype.update",
  "params": {
    "triggerid": "13938",
    "status": 0
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "13938"
    ]
  },
  "id": 1
}
```

Substituindo etiquetas de protótipo de trigger

Substituir etiquetas para um protótipo de trigger.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "triggerprototype.update",
  "params": {
    "triggerid": "17373",
    "tags": [
      {
        "tag": "volume",

```



```

        "value": "{#FSNAME}"
    },
    {
        "tag": "type",
        "value": "{#FSTYPE}"
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "triggerids": [
            "17373"
        ]
    },
    "id": 1
}

```

Fonte

CTriggerPrototype::update() in ui/include/classes/api/services/CTriggerPrototype.php.

Proxy

Esta classe foi projetada para trabalhar com proxies.

Referências de objetos:

- [Proxy](#)
- [Proxy interface](#)

Métodos disponíveis:

- [proxy.create](#) - criar novos proxies
- [proxy.delete](#) - excluir proxies
- [proxy.get](#) - recuperar proxies
- [proxy.update](#) - atualizar proxies

> Objeto proxy

Os objetos a seguir estão diretamente relacionados à API proxy.

Proxy

O objeto proxy tem as seguintes propriedades.

Propriedade	Tipo	Descrição
proxyid	string	(somente leitura) ID do proxy.
host (requerido)	string	Nome do proxy.
status (requerido)	integer	Tipo de proxy. Valores possíveis: 5 - proxy ativo; 6 - proxy passivo.
description	text	Descrição do proxy.

Propriedade	Tipo	Descrição
lastaccess	timestamp	(somente leitura) Hora em que o proxy se conectou pela última vez ao servidor.
tls_connect	integer	Conexões ao host. Os valores possíveis são: 1 - (padrão) Sem criptografia; 2 - PSK; 4 - certificado.
tls_accept	integer	Connections from host. Os valores de bitmap possíveis são: 1 - (padrão) Sem criptografia; 2 - PSK; 4 - certificado.
tls_issuer	string	Emissor do certificado.
tls_subject	string	Assunto do certificado.
tls_psk_identity	string	(somente escrita) identidade PSK. Obrigatório se <code>tls_connect</code> ou <code>tls_accept</code> tiver o PSK ativado. Não coloque informações confidenciais na identidade do PSK, elas são transmitidas sem criptografia pela rede para informar ao receptor qual PSK usar.
tls_psk	string	(somente escrita) A chave pré-compartilhada, com pelo menos 32 dígitos hexadecimais. Obrigatório se <code>tls_connect</code> ou <code>tls_accept</code> tiver o PSK ativado.
proxy_address	string	Endereços IP delimitados por vírgulas ou nomes DNS do proxy Zabbix ativo.
auto_compress	integer	(somente leitura) Indica se a comunicação entre o Zabbix Server e o proxy está compactada. Os valores possíveis são: 0 - Sem compactação; 1 - Compressão ativada;

Interface proxy

O objeto de interface proxy define a interface usada para se conectar a um proxy passivo. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
dns (requerido)	string	Nome DNS ao qual se conectar.
ip (requerido)	string	Pode estar vazio se as conexões forem feitas via endereço IP. Endereço IP ao qual se conectar.
port (requerido)	string	Pode estar vazio se as conexões forem feitas por meio de nomes DNS. Número da porta para se conectar.
useip (requerido)	integer	Se a conexão deve ser feita via endereço IP. Os valores possíveis são: 0 - conectar usando o nome DNS; 1 - conectar usando o endereço IP.

proxy.create

Descrição

```
object proxy.create(object/array proxies)
```

Este método permite criar novos proxies.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Veja [Usuário roles](#) para obter mais informações.

Parâmetros

(object/array) Proxies para criar.

Além das **propriedades de proxy padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
hosts	array	Hosts a serem monitorados pelo proxy. Se um host já for monitorado por um proxy diferente, ele será reatribuído ao proxy atual.
interface	object	Os hosts devem ter a propriedade <code>hostid</code> definida. Host interface a ser criado para o proxy passivo.
		Obrigatório para proxies passivos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos proxies criados na propriedade `proxyids`. A ordem dos IDs retornados corresponde à ordem dos proxies passados.

Exemplos**Criar um proxy ativo**

Crie um proxy de ação "Proxy ativo" e atribua um host para ser monitorado por ele.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.create",
  "params": {
    "host": "Active proxy",
    "status": "5",
    "hosts": [
      {
        "hostid": "10279"
      }
    ]
  },
  "auth": "ab9638041ec6922cb14b07982b268f47",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "proxyids": [
      "10280"
    ]
  },
  "id": 1
}
```

Criar um proxy passivo

Crie um proxy passivo "Proxy passivo" e atribua dois hosts para serem monitorados por ele.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.create",
  "params": {
    "host": "Passive proxy",
    "status": "6",
    "interface": {
      "ip": "127.0.0.1",
      "dns": "",
      "useip": "1",
      "port": "10051"
    },
    "hosts": [
      {
        "hostid": "10192"
      },
      {
        "hostid": "10139"
      }
    ]
  },
  "auth": "ab9638041ec6922cb14b07982b268f47",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "proxyids": [
      "10284"
    ]
  },
  "id": 1
}
```

Veja também

- [Host](#)
- [Proxy interface](#)

Fonte

CProxy::create() in ui/include/classes/api/services/CProxy.php.

proxy.delete

Descrição

object proxy.delete(array proxies)

Este método permite excluir proxies

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs de proxies a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos proxies excluídos na propriedade proxyids.

Exemplos

Excluir vários proxies

Exclua dois proxies.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.delete",
  "params": [
    "10286",
    "10285"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "proxyids": [
      "10286",
      "10285"
    ]
  },
  "id": 1
}
```

Fonte

CProxy::delete() in ui/include/classes/api/services/CProxy.php.

proxy.get

frontend### Descrição

integer/array proxy.get(object parameters)

O método permite recuperar proxies de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
proxyids	string/array	Retornar apenas proxies com os IDs fornecidos.
selectHosts	query	Retorna uma propriedade hosts com os hosts monitorados pelo proxy.
selectInterface	query	Retornar uma propriedade interface com a interface de proxy usada por um proxy passivo.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: hostid , host e status . Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	

Parâmetro	Tipo	Descrição
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recupere todos os proxies

Recupere todos os proxies configurados e suas interfaces.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.get",
  "params": {
    "output": "extend",
    "selectInterface": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "host": "Active proxy",
      "status": "5",
      "lastaccess": "0",
      "description": "",
      "tls_connect": "1",
      "tls_accept": "1",
      "tls_issuer": "",
      "tls_subject": "",
      "proxy_address": "",
      "auto_compress": "0",
      "proxyid": "30091",
      "interface": []
    },
    {
      "host": "Passive proxy",
      "status": "6",
      "lastaccess": "0",
      "description": "",
      "tls_connect": "1",
      "tls_accept": "1",
      "tls_issuer": "",
      "tls_subject": "",
      "proxy_address": "",

```

```

        "auto_compress": "0",
        "proxyid": "30092",
        "interface": {
            "interfaceid": "30109",
            "hostid": "30092",
            "useip": "1",
            "ip": "127.0.0.1",
            "dns": "",
            "port": "10051"
        }
    ],
    "id": 1
}

```

Veja também

- [Host](#)
- [Interface proxy](#)

Fonte

CProxy::get() in ui/include/classes/api/services/CProxy.php.

proxy.update

Descrição

object proxy.update(object/array proxies)

Este método permite atualizar proxies existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Veja [Usuário roles](#) para obter mais informações.

Parâmetros

(object/array) Propriedades de proxy a serem atualizadas.

A propriedade proxyid deve ser definida para cada proxy, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de proxy padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
hosts	array	Hosts a serem monitorados pelo proxy. Se um host já for monitorado por um proxy diferente, ele será reatribuído ao proxy atual.
interface	object	Os hosts devem ter a propriedade hostid definida. Host interface para substituir a interface existente para o proxy passivo.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos proxies atualizados na propriedade proxyids.

Exemplos

Alterar hosts monitorados por um proxy

Atualize o proxy para monitorar os dois hosts fornecidos.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.update",
  "params": {
    "proxyid": "10293",
    "hosts": [
      {
        "hostid": "10294"
      },
      {
        "hostid": "10295"
      },
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "proxyids": [
      "10293"
    ]
  },
  "id": 1
}
```

Alterar o status do proxy

Altere o proxy para um proxy ativo e renomeie-o para "Proxy ativo".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "proxy.update",
  "params": {
    "proxyid": "10293",
    "host": "Active proxy",
    "status": "5"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "proxyids": [
      "10293"
    ]
  },
  "id": 1
}
```

Veja também

- [Host](#)
- [Interface proxy](#)

Fonte

CProxy::update() in ui/include/classes/api/services/CProxy.php.

Registro de auditoria

Esta classe foi projetada para trabalhar com log de auditoria.

Referências de objetos:

- [objeto de log de auditoria](#)

Métodos disponíveis:

- [auditlog.get](#) - recupera auditoria registros de log

> Objeto de log de auditoria

Os objetos a seguir estão diretamente relacionados à API `auditlog`.

Registro de auditoria

O objeto de log de auditoria contém informações sobre as ações do usuário. Tem o seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----| |auditid|string|(readonly) ID da entrada do log de auditoria. Gerado usando o algoritmo CUID.| |userid|string|ID de usuário do autor da entrada de log de auditoria.| |username|string|Nome de usuário do autor da entrada do log de auditoria.| |clock|timestamp|Timestamp de criação da entrada do log de auditoria.| |ip|string|Endereço IP do autor da entrada do log de auditoria.| |ação|inteiro|Ação de entrada de registro de auditoria.

Os valores possíveis são:
0 - Adicionar;
1 - Atualizar;
2 - Excluir;
4 - Sair;
7 - Executar;
8 - Login;
9 - Falha no login;
10 - Histórico limpo.| |resourcetype|integer|Tipo de recurso de entrada de log de auditoria.

Os valores possíveis são:
0 - Usuário;
3 - Tipo de mídia;
4 - Host;
5 - Ação;
6 - Gráfico;
11 - Grupo de usuários;
13 - Acionador;
14 - Grupo de hosts;
15 - Item;
16 - Imagem;
17 - Valor map;
18 - Serviço;
19 - Mapa;
22 - Cenário da Web;
23 - Regra de descoberta;
25 - Script;
26 - Proxy;
27 - Manutenção;
28 - Expressão regular;
29 - Macro;
30 - Modelo;
31 - Protótipo de gatilho;
32 - Mapeamento de ícones;
33 - Painel;
34 - Correlação de eventos;
35 - Protótipo de gráfico;
36 - Protótipo de item;
37 - Protótipo de host;
38 - Registro automático;
39 - Módulo;
40 - Configurações;
41 - Limpeza;
42 - Autenticação;
43 - Painel de modelo;
44 - Função do usuário;
45 - Token de autenticação;
46 - Relatório agendado.| |resourceid|string|Identificador de recurso de entrada de log de auditoria.| |resourcenam|string|Nome legível do recurso de entrada de log de auditoria.| |recordsetid|string|ID do conjunto de registros da entrada do log de auditoria. Os registros de log de auditoria criados durante a mesma operação terão o mesmo ID do conjunto de registros. Gerado usando o algoritmo CUID.| |detalhes|texto|Detalhes da entrada do log de auditoria. Os detalhes são armazenados como objeto JSON onde cada nome de propriedade é um caminho para a propriedade ou objeto aninhado no qual ocorreu a alteração, e cada valor contém os dados sobre a alteração dessa propriedade em formato de matriz.

Os formatos de valor possíveis são :
["add"] - O objeto aninhado foi adicionado;
["add", "<value>"] - A propriedade do objeto adicionado contém <value>;
["update"] - O objeto aninhado foi atualizado;
["update", "<new value>", "<old value>"] - O valor da propriedade do objeto atualizado foi alterado de <old value> para <new value>;
["delete"] - O objeto aninhado foi excluído.

auditlog.get

Descrição

`integer/array auditlog.get(object parameters)`

O método permite recuperar registros de log de auditoria de acordo com o dado parâmetros.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----| |auditids|string/array|Retorna apenas o log de auditoria com os IDs fornecidos.| |userids|string/array|Retorna apenas o log de auditoria que foi criado pelos usuários fornecidos.| |time_from|timestamp|Retorna apenas as entradas de log de auditoria que foram criadas após ou em determinado momento.| |time_till|timestamp|Retorna apenas as entradas de log de auditoria que foram criadas antes ou em determinado momento.

|sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: auditid, userid, clock.| |filter|object|Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.

Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um valor único ou uma matriz de valores para correspondência.

Suporta adicionalmente a filtragem por campos de propriedade de detalhes: table_name, field_name.| |pesquisa|objeto|Pesquisa de substring sem distinção entre maiúsculas e minúsculas no conteúdo dos campos: username, ip, resourcename, details.| |countOutput|boolean|Esses parâmetros sendo comuns para todos os métodos get são descritos no [comentário de referência](#).| |excludePesquisa|boolean|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar log de auditoria

Recupera os dois últimos registros do log de auditoria.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "auditlog.get",
  "params": {
    "output": "extend",
    "sortfield": "clock",
    "sortorder": "DESC",
    "limit": 2
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "auditid": "cksstgfam0001yhdcc41y20q2",
      "userid": "1",
      "username": "Admin",
      "clock": "1629975715",
      "ip": "127.0.0.1",
      "action": "1",
      "resourcetype": "0",
      "resourceid": "0",
      "resourcename": "Jim",
      "recordsetid": "cksstgfal0000yhdcs067ondl",
      "details": "{\"user.name\": [\"update\", \"Jim\", \"\"], \"user.medias[37]\": [\"add\"], \"user.medias[38]\": [\"add\"]}"
    },
    {
      "auditid": "ckssofl0p0001yhdcqxclsg8r",
      "userid": "1",
      "username": "Admin",
      "clock": "1629967278",
      "ip": "127.0.0.1",
      "action": "0",
      "resourcetype": "0",
      "resourceid": "20",
      "resourcename": "John",
      "recordsetid": "ckssofl0p0000yhdcpxyo1jgo",
      "details": "{\"user.username\": [\"add\", \"John\"], \"user.userid\": [\"add\", \"20\"], \"user.username\": [\"add\", \"John\"]}"
    }
  ]
}
```

```

    }
  ],
  "id": 1
}

```

Veja também

- [objeto de log de auditoria](#)

Fonte

CAuditLog::get() em ui/include/classes/api/services/CAuditLog.php.

regra de descoberta

Esta classe foi projetada para trabalhar com regras de descoberta de rede.

Esta API deve funcionar com regras de descoberta de rede. Para as regras de descoberta de baixo nível, consulte a [regra LLD API](#).

Referências de objetos:

- [Regra de descoberta](#)

Métodos disponíveis:

- **drule.create** - criar novo regras de descoberta
- **drule.delete** - delete regras de descoberta
- **drule.get** - recupera descoberta as regras
- **drule.update** - atualização regras de descoberta

> Objeto de regra de descoberta

Os objetos a seguir estão diretamente relacionados à API drule.

Regra de descoberta

O objeto de regra de descoberta define uma regra de descoberta de rede. Tem o seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----| |druleid|string|(readonly) ID da regra de descoberta. | **iprange**
(obrigatório)|string|Um ou vários intervalos de IP a serem verificados separados por vírgulas.

Consulte a [configuração de descoberta de rede](#) seção para obter mais informações sobre formatos suportados de intervalos de IP. | **nome**
(obrigatório)|string|Nome da regra de descoberta. | **delay**|string|Intervalo de execução da regra de descoberta. Aceita segundos, unidade de tempo com sufixo e macro do usuário.

Padrão: 1h. | **nextcheck**|timestamp|(readonly) Hora em que a regra de descoberta será executada em seguida. | **proxy_hostid**|string|ID do proxy usado para descoberta. | **status**|inteiro|Se a regra de descoberta está habilitada.

Valores possíveis:
0 - (padrão) habilitado;
1 - desabilitado. |

drule.create

Descrição

```
object drule.create(object/array discoveryRules)
```

Este método permite criar novas regras de descoberta.

Note:

Este método está disponível apenas para os usuários do tipo Admin e Superadministrador. As permissões para chamar o método podem ser revogadas nas definições das funções de usuário (user roles) . Consulte [Usuário funções](#) para maiores informações.

Parâmetros

(object/array) Regras de descoberta a serem criadas.

Além da [propriedades padrão da regra de descoberta] (object#discovery_rule), o método aceita os seguintes parâmetros

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----|-----| **dchecks**
(obrigatório)|array|Discovery checks para criar para a regra de descoberta.]

Valores de retorno

(object) Retorna um objeto contendo os IDs das regras de descoberta sob a propriedade druleids. A ordem dos IDs retornados correspondem à ordem das regras de descoberta passadas.

Exemplos

Crie uma regra de descoberta

Crie uma regra de descoberta para encontrar máquinas executando o agente Zabbix na rede local. A regra deve usar uma única verificação do agente Zabbix na porta 10050.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "drule.create",
  "parâmetros": {
    "name": "Descoberta do agente Zabbix",
    "iprange": "192.168.1.1-255",
    "dchecks": [
      {
        "tipo": "9",
        "key_": "system.uname",
        "portas": "10050",
        "único": "0"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "druleids": [
      "6"
    ]
  },
  "id": 1
}
```

Veja também

- **Discovery check**

Fonte

CDRule::create() in ui/include/classes/api/services/CDRule.php.

drule.delete

Descrição

objeto drule.delete(array discoveryRuleIds)

Esse método permite remover regras de descoberta.

Note:

Esse método está disponível somente para usuários Admin e Super admin. As permissões para invocar o método podem ser revogadas nas configurações de usuário. Veja **Usuário função** para mais informações.

Parâmetros

(array) IDs das regras de descoberta a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs da descoberta excluída regras sob a propriedade `druleids`.

Exemplos

Excluir várias regras de descoberta

Exclua duas regras de descoberta.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "drule.delete",
  "parâmetros": [
    "4",
    "6"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "druleids": [
      "4",
      "6"
    ]
  },
  "id": 1
}
```

Fonte

`CDRule::delete()` in `ui/include/classes/api/services/CDRule.php`.

`drule.get`

Descrição

`integer/array drule.get(object parameters)`

O método permite recuperar as regras de descoberta de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| |dhostids|string/array|Retorna apenas as regras de descoberta que criaram os hosts descobertos fornecidos.| |druleids|string/array|Retorna apenas regras de descoberta com os IDs fornecidos.| |dserviceids|string/array|Retorna apenas as regras de descoberta que criaram os serviços descobertos fornecidos.| |selectDChecks|query|Retorne uma propriedade **dchecks** com as verificações de descoberta usadas pela regra de descoberta.

Suporta count.| |selectDHosts|query|Retorne uma propriedade **dhosts** com os hosts descobertos criados pela regra de descoberta.

Suporta count.| |limitSelects|integer|Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:

selectDChecks - os resultados serão classificados por **dcheckid**;

selectDHosts - resultados será ordenado por **dhostsid**.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: **druleid** e **name**.| |countOutput|boolean|Esses parâmetros

sendo comuns para todos os métodos get são descritos em detalhes no [comentário de referência](#).| |editável|booleano|^| |excluiPesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recupere todas as regras de descoberta

Recupere todas as regras de descoberta configuradas e a descoberta as verifica usar.

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "drule.get",
  "parâmetros": {
    "saída": "estender",
    "selectDChecks": "estender"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "druleid": "2",
      "proxy_hostid": "0",
      "nome": "Rede local",
      "iprange": "192.168.3.1-255",
      "atraso": "5s",
      "nextcheck": "1348754327",
      "estado": "0",
      "dchecks": [
        {
          "dcheckid": "7",
          "druleid": "2",
          "tipo": "3",
          "chave_": "",
          "snmp_community": "",
          "portas": "21",
          "snmpv3_securityname": "",
          "snmpv3_securitylevel": "0",
          "snmpv3_authpassphrase": "",
          "snmpv3_privpassphrase": "",
          "único": "0",
          "snmpv3_authprotocol": "0",
          "snmpv3_privprotocol": "0",
          "host_source": "1",
          "name_source": "0"
        },
        {
          "dcheckid": "8",
          "druleid": "2",
          "tipo": "4",
          "chave_": "",
          "snmp_community": "",
          "portas": "80",

```

```

        "snmpv3_securityname": "",
        "snmpv3_securitylevel": "0",
        "snmpv3_authpassphrase": "",
        "snmpv3_privpassphrase": "",
        "único": "0",
        "snmpv3_authprotocol": "0",
        "snmpv3_privprotocol": "0",
        "host_source": "1",
        "name_source": "0"
    }
]
},
{
    "druleid": "6",
    "proxy_hostid": "0",
    "name": "Descoberta do agente Zabbix",
    "iprange": "192.168.1.1-255",
    "atraso": "1h",
    "nextcheck": "0",
    "estado": "0",
    "dchecks": [
        {
            "dcheckid": "10",
            "druleid": "6",
            "tipo": "9",
            "key_": "system.uptime",
            "snmp_community": "",
            "portas": "10050",
            "snmpv3_securityname": "",
            "snmpv3_securitylevel": "0",
            "snmpv3_authpassphrase": "",
            "snmpv3_privpassphrase": "",
            "único": "0",
            "snmpv3_authprotocol": "0",
            "snmpv3_privprotocol": "0",
            "host_source": "2",
            "name_source": "3"
        }
    ]
}
],
"id": 1
}

```

Veja também

- [Descoberto host](#)
- [Descoberta check](#)

Fonte

CDRule::get() em ui/include/classes/api/services/CDRule.php.

drule.update

Descrição

object drule.update(object/array discoveryRules)

Este método permite atualizar as regras de descoberta existentes.

Note:

Este método está disponível apenas para Admin e Superadministrador tipos de usuário. As permissões para chamar o método podem ser revogadas na função de usuário definições. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object/array) Propriedades da regra de descoberta a serem atualizadas.

A propriedade `druleid` deve ser definida para cada regra de descoberta, todas outras propriedades são opcionais. Somente as propriedades passadas serão atualizado, todos os outros permanecerão inalterados.

Além da **regra de descoberta padrão properties**, o método aceita o seguinte parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----| |dchecks|array|Discovery **checks** para substituir as verificações existentes.]

Valores de retorno

(object) Retorna um objeto contendo os IDs da descoberta atualizada regras sob a propriedade `druleids`.

Exemplos

Alterar o intervalo de IP de uma regra de descoberta

Altere o intervalo de IP de uma regra de descoberta para "192.168.2.1-255".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "drule.update",
  "parâmetros": {
    "druleid": "6",
    "iprange": "192.168.2.1-255"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "druleids": [
      "6"
    ]
  },
  "id": 1
}
```

Veja também

- [Descoberta check](#)

Fonte

`CDRule::update()` em `ui/include/classes/api/services/CDRule.php`.

Regra LLD

Esta classe foi projetada para trabalhar com regras de descoberta de baixo nível.

Referências de objetos:

- [Regra LLD](#)

Métodos disponíveis:

- [discoveryrule.copy](#) - copiando regras LLD

- **discoveryrule.create** - criando novas regras LLD
- **discoveryrule.delete** - excluindo regras LLD
- **discoveryrule.get** - recuperando regras LLD
- **discoveryrule.update** - atualizando as regras LLD

> Regra de objeto LLD

Os objetos a seguir estão diretamente relacionados à API **discoveryrule**.

Regra LLD

O objeto de regra de descoberta de baixo nível tem as seguintes propriedades..

Propriedade	Tipo	Descrição
itemid	string	(readonly) ID of the LLD rule.
delay (requerido)	string	Update interval of the LLD rule. A Aceita segundos ou unidade de tempo com sufixo e com ou sem um ou mais intervalos personalizados que consistem em intervalos flexíveis e intervalos de agendamento como strings serializadas. Também aceita macros de usuário. Intervalos flexíveis podem ser escritos como duas macros separadas por uma barra. Os intervalos são separados por um ponto e vírgula.
hostid (requerido)	string	Opcional para o Zabbix trapper, itens dependentes e para o Zabbix agent(ativo) com a chave <code>mqtt.get</code> . ID do host ao qual a regra LLD pertence.
interfaceid (requerido)	string	ID da interface do host da regra LLD. Usado apenas para regras LLD do host.
key_ (requerido)	string	Não é necessário para o agente Zabbix (ativo), Zabbix internal, Zabbix trapper, dependente, dependente, monitor de banco de dados e script de regras LLD. Opcional para regras LLD do agente HTTP. Key da regra LLD.
name (requerido)	string	Nome da regra LLD.
type (requerido)	integer	Tipo de regra LLD. Valores possíveis: 0 - Zabbix agent; 2 - Zabbix trapper; 3 - simple check; 5 - Zabbix internal; 7 - Zabbix agent (ativo); 10 - external check; 11 - database monitor; 12 - IPMI agent; 13 - SSH agent; 14 - TELNET agent; 16 - JMX agent; 18 - Dependent item; 19 - HTTP agent; 20 - SNMP agent; 21 - Script.
url (requerido)	string	String de URL, necessária para a regra LLD do agente HTTP. Suporta macros de usuário, {HOST.IP}, {HOST.CONN}, {HOST.DNS}, {HOST.HOST}, {HOST.NAME}, {ITEM.ID}, {ITEM.KEY}.
allow_traps	integer	Campo de regra LLD do agente HTTP. Permitir preencher o valor como no tipo de item do trapper também. 0 - (default) Não permite aceitar dados recebidos. 1 - Permite aceitar dados recebidos.

Propriedade	Tipo	Descrição
authtype	integer	Usado apenas pelo SSH agent ou regras LLD do agente HTTP. Valores possíveis do método de autenticação do SSH agent: 0 - (default) senha; 1 - chave pública. HTTP Valores possíveis do método de autenticação do agente HTTP: 0 - (default) Nenhum 1 - básico 2 - NTLM
description	string	Descrição da regra LLD.
error	string	(somente leitura) Mensagem de erro se houver problemas ao atualizar a regra LLD.
follow_redirects	integer	Campo de regra LLD do agente HTTP. Siga os redirecionamentos de resposta ao agrupar dados. 0 - Não siga redirecionamentos. 1 - (default) Seguir redirecionamentos.
headers	object	Campo de regra LLD do agente HTTP. Objeto com cabeçalhos de solicitação HTTP(S), em que o nome do cabeçalho é usado como chave e o valor do cabeçalho como valor. Exemplo: { "User-Agent": "Zabbix" }
http_proxy	string	Campo de regra LLD do agente HTTP. Cadeia de conexão do proxy HTTP(S).
ipmi_sensor	string	Sensor IPMI. Usado apenas pelas regras do IPMI LLD.
jmx_endpoint	string	Cadeia de conexão personalizada do agente JMX.
lifetime	string	Valor padrão: service:jmx:rmi:///jndi/rmi://{HOST.CONN}:{HOST.PORT}/jmxrmi Período de tempo após o qual os itens que não são mais descobertos serão excluídos. Aceita segundos, unidade de tempo com sufixo e macro do usuário.
master_itemid	integer	Padrão: 30d. ID do item mestre. Recursão de até 3 itens dependentes e contagem máxima de itens dependentes igual a 999 são permitidas. A regra de descoberta não pode ser um item mestre para outra regra de descoberta.
output_format	integer	Obrigatório para item dependente. Campo de regra LLD do agente HTTP. A resposta deve ser convertida em JSON. 0 - (padrão) Armazenamento não tratado. 1 - Converter para JSON.
params	string	Parâmetros adicionais dependendo do tipo da regra LLD: - script executado para regras SSH e Telnet LLD; - Consulta SQL para regras LLD do monitor de banco de dados; - fórmula para regras LLD calculadas.
parameters	array	Parâmetros adicionais para a regra LLD do tipo script. Matriz de objetos com propriedades 'name' e 'value', onde o nome deve ser único.
password	string	Senha para autenticação. Usado por verificação simples, SSH, Telnet, monitor de banco de dados, JMX e regras LLD do agente HTTP.
post_type	integer	Campo de regra LLD do agente HTTP. Tipo de corpo de dados de postagem armazenado na propriedade de postagen 0 - (padrão) Dados não tratados. 2 - Dados JSON. 3 - Dados XML.

Propriedade	Tipo	Descrição
posts	string	Campo de regra LLD do agente HTTP. Dados do corpo da solicitação HTTP(S). Usado com post_type.
privatekey	string	Nome do arquivo de chave privada.
publickey	string	Nome do arquivo de chave pública.
query_fields	array	Campo de regra LLD do agente HTTP. Parâmetros de consulta. Matriz de objetos com pares 'key': 'value' onde valor pode ser uma string vazia.
request_method	integer	Campo de regra LLD do agente HTTP. Tipo de método de solicitação. 0 - (padrão) GET 1 - POST 2 - PUT 3 - HEAD
retrieve_mode	integer	Campo de regra LLD do agente HTTP. Que parte da resposta deve ser armazenada. 0 - (padrão) Body. 1 - Headers. 2 - O corpo e os cabeçalhos serão armazenados.
snmp_oid	string	Para request_method HEAD apenas 1 é o valor permitido. SNMP OID.
ssl_cert_file	string	Campo de regra LLD do agente HTTP. Caminho do arquivo de chave SSL pública.
ssl_key_file	string	Campo de regra LLD do agente HTTP. Caminho do arquivo de chave SSL privada.
ssl_key_password	string	Campo de regra LLD do agente HTTP. Senha para arquivo de chave SSL.
state	integer	(somente leitura) Estado da regra LLD. Valores possíveis: 0 - (padrão) normal; 1 - não suportado.
status	integer	Status da regra LLD. Valores possíveis: 0 - (padrão) regra LLD habilitada; 1 - regra LLD desabilitada.
status_codes	string	Campo de regra LLD do agente HTTP. Intervalos de códigos de status HTTP necessários separados por vírgulas. Também suporta macros de usuário como parte da lista separada por vírgulas.
templateid	string	Exemplo: 200,200-{\$M},{M},200-400 (somente leitura) ID da regra LLD do modelo pai.
timeout	string	Tempo limite da solicitação de sondagem de dados do item. Usado para regras de agente HTTP e script LLD. Suporta macros de usuário. padrão: 3s valor máximo: 60s
trapper_hosts	string	Hosts permitidos. Usado por regras LLD do trapper ou regras LLD do agente HTTP.
username	string	Username for authentication. Used by simple check, SSH, Telnet, database monitor, JMX and HTTP agent LLD rulesNome de usuário para autenticação. Usado por verificação simples, SSH, Telnet, monitor de banco de dados, JMX e regras LLD do agente HTTP.
uuid	string	Requerido pelas regras SSH e Telnet LLD. Identificador único universal, usado para vincular regras LLD importadas a regras já existentes. Usado apenas para regras LLD em modelos. Gerado automaticamente, se não fornecido.
Para operações de atualização este campo é somente leitura.		

Propriedade	Tipo	Descrição
verify_host	integer	Campo de regra LLD do agente HTTP. Validar o nome do host na URL está no campo Common Name ou no campo Subject Alternate Name do certificado do host.
verify_peer	integer	Campo de regra LLD do agente HTTP. Validar é o certificado do host autêntico.

Filtro de regra LLD

O objeto de filtro de regras LLD define um conjunto de condições que podem ser usadas para filtrar objetos descobertos. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
conditions (requerido)	array	Conjunto de condições de filtro a serem usadas para filtrar resultados.
evaltype (requerido)	integer	Método de avaliação da condição do filtro.
		Valores possíveis: 0 - and/or; 1 - and; 2 - or; 3 - expressão personalizada.
eval_formula	string	(somente leitura) Expressão gerada que será usada para avaliar as condições do filtro. A expressão contém IDs que fazem referência a condições de filtro específicas por seu formulaid. O valor de eval_formula é igual ao valor de formula para filtros com uma expressão personalizada.
formula	string	Expressão definida pelo usuário a ser usada para avaliar condições de filtros com uma expressão personalizada. A expressão deve conter IDs que façam referência a condições de filtro específicas por seu formulaid. Os IDs usados na expressão devem corresponder exatamente aos definidos nas condições do filtro: nenhuma condição pode permanecer não utilizada ou omitida.
		Obrigatório para filtros de expressão personalizados.

Condição do filtro de regra LLD

O objeto de condição de filtro de regra LLD define uma verificação separada a ser executada no valor de uma macro LLD. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
macro (requerido)	string	Macro LLD para realizar a verificação.
value (requerido)	string	Valor para comparar.
formulaid	string	ID exclusivo arbitrário que é usado para fazer referência à condição de uma expressão personalizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente.

Propriedade	Tipo	Descrição
operator	integer	Operador de condição. Valores possíveis: 8 - (padrão) corresponde à expressão regular; 9 - não corresponde à expressão regular; 12 - existe; 13 - não existe.

Note:

Para entender melhor como usar filtros com vários tipos de expressões, veja exemplos nas páginas de método [discoveryrule.get](#) e [discoveryrule.create](#).

Caminho da macro LLD

O caminho da macro LLD tem as seguintes propriedades:

Propriedade	Tipo	Descrição
lld_macro (requerido)	string	Macro LLD.
path (requerido)	string	Seletor para valor que será atribuído à macro correspondente.

Pré-processamento de regra LLD

O objeto de pré-processamento de regra LLD tem as seguintes propriedades.

Propriedade	Tipo	Descrição
type (requerido)	integer	O tipo de opção de pré-processamento. Valores possíveis: 5 - Correspondência de expressão regular; 11 - XML XPath; 12 - JSONPath; 15 - Não corresponde a expressão regular; 16 - Verificar erro no JSON; 17 - Verificar erro no XML; 20 - Descartar inalterado com pulsação; 23 - Prometheus para JSON; 24 - CSV para JSON; 25 - Substituir; 27 - XML para JSON.
params (requerido)	string	Additional Parâmetros adicionais usados pela opção de pré-processamento. Vários parâmetros são separados pelo caractere LF (\n).
error_handler (requerido)	integer	Tipo de ação usado em caso de falha na etapa de pré-processamento. Valores possíveis: 0 - Mensagem de erro é definida pelo Zabbix server; 1 - Descartar valor; 2 - Definir valor personalizado;< br>3 - Definir mensagem de erro personalizada.
error_handler_params (requerido)	string	Parâmetros do manipulador de erros. Usado com error_handler . Deve estar vazio, se error_handler for 0 ou 1. Pode estar vazio se, error_handler for 2. Não pode estar vazio, se error_handler é 3.

Os seguintes parâmetros e manipuladores de erros são suportados para cada tipo de pré-processamento.

Tipo de pré-processamento	Nome	Parâmetro 1	Parâmetro 2	Parâmetro 3	Manipuladores de erros compatíveis
5	Expressão regular	pattern ¹	output ²		0, 1, 2, 3
11	XML XPath	path ³			0, 1, 2, 3
12	JSONPath	path ³			0, 1, 2, 3
15	Não corresponde à expressão regular	pattern ¹			0, 1, 2, 3
16	Verifique se há erro no JSON	path ³			0, 1, 2, 3
17	Verifique se há erro no XML	path ³			0, 1, 2, 3
20	Descarte inalterado com heart-beat	seconds ^{4, 5, 6}			
23	Prometheus para JSON	pattern ^{5, 7}			0, 1, 2, 3
24	CSV para JSON	character ²	character ²	0,1	0, 1, 2, 3
25	Substitua	search string ²	replacement ²		
27	XML para JSON				0, 1, 2, 3

¹ expressão regular

² string

³ JSONPath ou XML XPath

⁴ inteiro positivo (com suporte de sufixos de tempo, ex. 30s, 1m, 2h, 1d)

⁵ macro de usuário

⁶ macro LLD

⁷ padrão do Prometheus seguindo a sintaxe: <metric name>{<label name>=<label value>", ...} == <value>. Cada componente padrão do Prometheus (métrica, nome do rótulo, valor do rótulo e valor da métrica) pode ser macro do usuário.

⁸ Saída do Prometheus seguindo a sintaxe: <label name>.

Substituições de regras LLD

O objeto de substituição de regra LLD define um conjunto de regras (filtros, condições e operações) que são usadas para substituir propriedades de diferentes objetos de protótipo. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
name (requerido)	string	Nome de substituição exclusivo.
step (requerido)	integer	Número de pedido exclusivo da substituição.
stop	integer	Parar de processar as próximas substituições se corresponder.
		Valores possíveis: 0 - (padrão) não pare de processar substituições; 1 - pare de processar substituições se o filtro corresponder.
filter	object	Substituir filtro.
operations	array	Substituir operações.

Filtro de substituição de regra LLD

O objeto de filtro de substituição de regra LLD define um conjunto de condições que, se corresponderem ao objeto descoberto, a substituição será aplicada. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
evaltype (requerido)	integer	Substituir o método de avaliação da condição do filtro. Valores possíveis: 0 - e/ou; 1 - e; 2 - ou; 3 - expressão personalizada.
conditions (requerido)	array	Conjunto de condições de filtro de substituição a serem usadas para corresponder aos objetos descobertos.
eval_formula	string	(somente leitura) Expressão gerada que será usada para avaliar as condições do filtro de substituição. A expressão contém códigos que fazem referência a condições de filtro de substituição específicas por seu formulaid. O valor de eval_formula é igual ao valor de formula para filtros com uma expressão personalizada.
formula	string	Expressão definida pelo usuário a ser usada para avaliar condições de filtros de substituição com uma expressão personalizada. A expressão deve conter IDs que façam referência a condições de filtro de substituição específicas por seu formulaid. Os IDs usados na expressão devem corresponder exatamente aos definidos nas condições do filtro de substituição: nenhuma condição pode permanecer sem uso ou omitida. Obrigatório para filtros de substituição de expressão personalizados.

Condição de filtro de substituição de regra LLD

O objeto de condição de filtro de substituição de regra LLD define uma verificação separada a ser executada no valor de uma macro LLD. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
macro (requerido)	string	Macro LLD para realizar a verificação.
value (requerido)	string	Valor para comparar.
formulaid	string	ID exclusivo arbitrário que é usado para fazer referência à condição de uma expressão personalizada. Só pode conter letras maiúsculas. O ID deve ser definido pelo usuário ao modificar as condições do filtro, mas será gerado novamente ao solicitá-los posteriormente.

Propriedade	Tipo	Descrição
operator	integer	Operador de condição. Valores possíveis: 8 - (padrão) corresponde à expressão regular; 9 - não corresponde à expressão regular; 12 - existe; 13 - corresponde não existe.

Operação de substituição de regra LLD

A operação de substituição da regra LLD é uma combinação de condições e ações a serem executadas no objeto protótipo. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
operationobject (requerido)	integer	Tipo de objeto descoberto para realizar a ação. Valores possíveis: 0 - Protótipo do item; 1 - Protótipo do trigger; 2 - Protótipo do gráfico; 3 - Protótipo do host.
operator	integer	Operador de condição de substituição. Valores possíveis: 0 - (padrão) é igual; 1 - não é igual; 2 - contém; 3 - não contém; 8 - corresponde; 9 - não corresponde.
value	string	Padrão para corresponder ao nome do item, trigger, gráfico ou protótipo do host, dependendo do objeto selecionado.
opstatus	object	Substituir objeto de status de operação para objetos de protótipo de item, trigger e host.
opdiscover	object	Operação de substituição descobrir objeto de status (todos os tipos de objeto).
opperiod	object	Substituir objeto de período de operação (intervalo de atualização) para objeto de protótipo de item.
ohistory	object	Substituir objeto de histórico de operação para objeto de protótipo de item.
optrends	object	Substituir objeto de tendências de operação para objeto protótipo de item.
opseverity	object	Substituir objeto de gravidade de operação para objeto protótipo de trigger.
optag	array	Substituir objeto de etiqueta de operação para objetos de protótipo de trigger e host.
optemplate	array	Substituir objeto de modelo de operação para objeto de protótipo de host.
opinventory	object	Substituir objeto de inventário de operação para objeto de protótipo do host.

Status da operação de substituição da regra LLD

A regra LLD substitui o status de operação definido como objeto descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
status (requerido)	integer	Substituir o status do objeto selecionado. Valores possíveis: 0 - Criar ativado; 1 - Criar desativado.

Descoberta de operação de substituição de regra LLD

O status de descoberta da operação de substituição da regra LLD é definido como objeto descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
discover (requerido)	integer	Substituir o status de descoberta do objeto selecionado. Valores possíveis: 0 - Sim, continue descobrindo os objetos; 1 - Não, novos objetos não serão descobertos e os existentes serão marcados como perdidos.

Período de operação de substituição da regra LLD

O período de operação de substituição de regra LLD é um valor de intervalo de atualização (suporta intervalos personalizados) que está definido como item descoberto. Tem o seguinte propriedades:

Propriedade	Tipo	Descrição
delay (obrigatório)	string	Substituir o intervalo de atualização do protótipo de item. Aceita segundos ou uma unidade de tempo com sufixo (30s,1m,2h,1d) bem como intervalos flexíveis e agendados e macros de usuário ou macros LLD. Vários intervalos são separados por um ponto e vírgula.

Histórico de operação de substituição de regra LLD

Valor do histórico da operação de substituição da regra LLD que está definido como item descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
histórico (required)	string	Substitua o histórico do protótipo do item, que é uma unidade de tempo de quanto tempo os dados do histórico devem ser armazenados. Também aceita macro de usuário e macro LLD.

Tendências de operação de substituição de regra LLD

Valor de tendências de operação de substituição de regra LLD que é definido como item descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
Tendências (obrigatório)	string	Substitua as tendências do protótipo do item, que é uma unidade de tempo de quanto tempo os dados de tendências devem ser armazenados. Também aceita macro de usuário e macro LLD.

Severidade da operação de substituição da regra LLD

O valor de gravidade da operação de substituição da regra LLD que está definido para o gatilho descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
Severidade (obrigatório)	inteiro	Substituir a gravidade do protótipo do acionador. Os valores possíveis são: 0 - (padrão) não classificado; 1 - informação; 2 - atenção; 3 - média; 4 - alta; 5 - desastre.

Etiqueta de operação de substituição de regra LLD

O objeto de etiqueta de operação de substituição de regra LLD contém o nome e o valor da etiqueta que são definidos como objeto descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
etiqueta (obrigatório)	string	Novo nome da etiqueta.
valor	string	Novo valor da etiqueta.

Modelo de operação de substituição de regra LLD

Objeto de modelo de operação de substituição de regra LLD que está vinculado ao host descoberto. Tem as seguintes propriedades:

Propriedade	Tipo	Descrição
Template ID (obrigatório)	string	Substitua o modelo de modelos vinculados de protótipo de host.

Inventário de operação de substituição de regra LLD

A regra LLD substitui o valor do modo de inventário da operação definido como host descoberto. Possui as seguintes propriedades:

Propriedade	Tipo	Descrição
inventory_mode (required)	integer	Substitua o modo de inventário do protótipo do host. Os valores possíveis são: -1 - desativado; 0 - (padrão) manual; 1 - automático.

discoveryrule.copy

Descrição

`object discoveryrule.copy(object parameters)`

Este método permite copiar as regras LLD com todos os protótipos para o hospedeiros dados.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem as regras LLD a serem copiadas e os hosts de destino.

Parameter	Type	Description
discoveryids	array	IDs of the LLD rules to be copied.
hostids	array	IDs of the hosts to copy the LLD rules to.

Valores de retorno

(boolean) Retorna true se a cópia foi bem sucedida.

Exemplos

Copie uma regra LLD para vários hosts

Copie uma regra LLD para dois hosts.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.copy",
  "params": {
    "discoveryids": [
      "27426"
    ],
    "hostids": [
      "10196",
      "10197"
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": true,
  "id": 1
}
```

Fonte

CDiscoveryrule::copy() in ui/include/classes/api/services/CDiscoveryRule.php.

discoveryrule.create

Descrição

object discoveryrule.create(object/array lldRules)

Este método permite criar novas regras LLD.

Note:

Este método está disponível apenas para os tipos de usuário Admin and Super admin As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object/array) Regras LLD para criar.

Além das [propriedades da regra LLD padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
filter	object	Objeto de filter objeto para a regra LLD
preprocessing	array	Opções do Regra LLD preprocessing .
lld_macro_paths	array	Opções do Regra LLD lld_macro_path .

Parâmetro	Tipo	Descrição
overrides	array	Opções do Regra LLD overrides .

Valores de retorno

(object) Retorna um objeto contendo os IDs das regras LLD criadas na propriedade `itemids`. A ordem dos IDs retornados corresponde à ordem das regras LLD passadas.

Exemplos

Criando uma regra LLD

Crie uma regra LLD do agente Zabbix para descobrir sistemas de arquivos montados. Os itens descobertos serão atualizados a cada 30 segundos.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "name": "Mounted filesystem discovery",
    "key_": "vfs.fs.discovery",
    "hostid": "10197",
    "type": 0,
    "interfaceid": "112",
    "delay": "30s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "27665"
    ]
  },
  "id": 1
}
```

Usando um filtro

Crie uma regra LLD com um conjunto de condições para filtrar os resultados. As condições serão agrupadas usando o operador lógico "and".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "name": "Filtered LLD rule",
    "key_": "lld",
    "hostid": "10116",
    "type": 0,
    "interfaceid": "13",
    "delay": "30s",
    "filter": {
      "evaltype": 1,
      "conditions": [
        {
          "macro": "#{MACRO1}",
          "value": "@regex1"
        }
      ]
    }
  }
}
```

```

        },
        {
            "macro": "#{MACRO2}",
            "value": "@regex2",
            "operator": "9"
        },
        {
            "macro": "#{MACRO3}",
            "value": "",
            "operator": "12"
        },
        {
            "macro": "#{MACRO4}",
            "value": "",
            "operator": "13"
        }
    ]
}
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "27665"
        ]
    },
    "id": 1
}

```

Criando uma regra LLD com caminhos de macro

Request:

```

{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "LLD rule with LLD macro paths",
        "key_": "lld",
        "hostid": "10116",
        "type": 0,
        "interfaceid": "13",
        "delay": "30s",
        "lld_macro_paths": [
            {
                "lld_macro": "#{MACRO1}",
                "path": "$.path.1"
            },
            {
                "lld_macro": "#{MACRO2}",
                "path": "$.path.2"
            }
        ]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "27665"
    ]
  },
  "id": 1
}
```

Usando um filtro de expressão personalizado

Crie uma regra LLD com um filtro que usará uma expressão personalizada para avaliar as condições. A regra LLD deve descobrir apenas objetos cujo valor de macro "{#MACRO1}" corresponde à expressão regular "regex1" e "regex2", e o valor de "{#MACRO2}" corresponde a "regex3" ou "regex4". Os IDs de fórmula "A", "B", "C" e "D" foram escolhidos arbitrariamente.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "name": "Filtered LLD rule",
    "key_": "lld",
    "hostid": "10116",
    "type": 0,
    "interfaceid": "13",
    "delay": "30s",
    "filter": {
      "evaltype": 3,
      "formula": "(A and B) and (C or D)",
      "conditions": [
        {
          "macro": "{#MACRO1}",
          "value": "@regex1",
          "formulaid": "A"
        },
        {
          "macro": "{#MACRO1}",
          "value": "@regex2",
          "formulaid": "B"
        },
        {
          "macro": "{#MACRO2}",
          "value": "@regex3",
          "formulaid": "C"
        },
        {
          "macro": "{#MACRO2}",
          "value": "@regex4",
          "formulaid": "D"
        }
      ]
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
```

```

        "27665"
    ],
    },
    "id": 1
}

```

Usando campos e cabeçalhos de consulta personalizados

Crie uma regra LLD com campos e cabeçalhos de consulta personalizados.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "hostid": "10257",
    "interfaceid": "5",
    "type": 19,
    "name": "API HTTP agent",
    "key_": "api_discovery_rule",
    "value_type": 3,
    "delay": "5s",
    "url": "http://127.0.0.1?discoverer.php",
    "query_fields": [
      {
        "mode": "json"
      },
      {
        "elements": "2"
      }
    ],
    "headers": {
      "X-Type": "api",
      "Authorization": "Bearer mF_A.B5f-2.1JcM"
    },
    "allow_traps": 1,
    "trapper_hosts": "127.0.0.1"
  },
  "auth": "d678e0b85688ce578ff061bd29a20d3b",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "28336"
    ]
  },
  "id": 1
}

```

Criando uma regra LLD com pré-processamento

Request:

```

{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "name": "Discovery rule with preprocessing",
    "key_": "lld.with.preprocessing",
    "hostid": "10001",
    "ruleid": "27665",

```

```

        "type": 0,
        "value_type": 3,
        "delay": "60s",
        "interfaceid": "1155",
        "preprocessing": [
            {
                "type": 20,
                "params": "20",
                "error_handler": 0,
                "error_handler_params": ""
            }
        ]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "itemids": [
            "44211"
        ]
    },
    "id": 1
}

```

Como criar uma regra LLD com substituições

Request:

```

{
    "jsonrpc": "2.0",
    "method": "discoveryrule.create",
    "params": {
        "name": "Discover database host",
        "key_": "lld.with.overrides",
        "hostid": "10001",
        "type": 0,
        "value_type": 3,
        "delay": "60s",
        "interfaceid": "1155",
        "overrides": [
            {
                "name": "Discover MySQL host",
                "step": "1",
                "stop": "1",
                "filter": {
                    "evaltype": "2",
                    "conditions": [
                        {
                            "macro": "{#UNIT.NAME}",
                            "operator": "8",
                            "value": "~mysqld\\.service$"
                        },
                        {
                            "macro": "{#UNIT.NAME}",
                            "operator": "8",
                            "value": "~mariadb\\.service$"
                        }
                    ]
                }
            }
        ]
    },
    "operations": [

```



```

        {
            "operationobject": "3",
            "operator": "2",
            "value": "Database host",
            "opstatus": {
                "status": "0"
            },
            "optemplate": [
                {
                    "templateid": "10170"
                }
            ],
            "optag": [
                {
                    "tag": "Database",
                    "value": "MySQL"
                }
            ]
        }
    ],
    {
        "name": "Discover PostgreSQL host",
        "step": "2",
        "stop": "1",
        "filter": {
            "evaltype": "0",
            "conditions": [
                {
                    "macro": "{#UNIT.NAME}",
                    "operator": "8",
                    "value": "^postgresql\\.service$"
                }
            ]
        },
        "operations": [
            {
                "operationobject": "3",
                "operator": "2",
                "value": "Database host",
                "opstatus": {
                    "status": "0"
                },
                "optemplate": [
                    {
                        "templateid": "10263"
                    }
                ],
                "optag": [
                    {
                        "tag": "Database",
                        "value": "PostgreSQL"
                    }
                ]
            }
        ]
    }
],
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "30980"
    ]
  },
  "id": 1
}
```

Criar regra LLD de script

Crie uma coleta de dados simples usando uma regra LLD de script.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.create",
  "params": {
    "name": "Script example",
    "key_": "custom.script.lldrule",
    "hostid": "12345",
    "type": 21,
    "value_type": 4,
    "params": "var request = new CurlHttpRequest();\nreturn request.Post(\"https://postman-echo.com/post\");",
    "parameters": [
      {
        "name": "host",
        "value": "{HOST.CONN}"
      }
    ],
    "timeout": "6s",
    "delay": "30s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 2
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "23865"
    ]
  },
  "id": 3
}
```

Veja também

- [LLD rule filter](#)
- [LLD macro paths](#)
- [LLD rule preprocessing](#)

Fonte

CDiscoveryRule::create() in ui/include/classes/api/services/CDiscoveryRule.php.

discoveryrule.delete

Descrição

object discoveryrule.delete(array lldRuleIds)

Este método permite excluir regras LLD.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(array) IDs das regras LLD a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das regras LLD excluídas na propriedade `itemids`.

Exemplos

Excluindo várias regras LLD

Excluir duas regras LLD.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.delete",
  "params": [
    "27665",
    "27668"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "ruleids": [
      "27665",
      "27668"
    ]
  },
  "id": 1
}
```

Fonte

`CDiscoveryRule::delete()` in `ui/include/classes/api/services/CDiscoveryRule.php`.

discoveryrule.get

Descrição

`integer/array discoveryrule.get(object parameters)`

O método permite recuperar as regras LLD de acordo com os parâmetros recebidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para invocar o método podem ser revogadas nas configurações de função do usuário. Veja [Usuário funções](#) para mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
itemids	string/array	Retorne apenas regras LLD com os IDs fornecidos.
groupids	string/array	Retornar apenas as regras LLD que pertencem aos hosts dos grupos fornecidos.
hostids	string/array	Retornar apenas as regras LLD que pertencem aos hosts fornecidos.
inherited	boolean	Se definido como <code>true</code> , retorna apenas as regras LLD herdadas de um template.
interfaceids	string/array	Retornar apenas as regras LLD usam as interfaces de host fornecidas.
monitored	boolean	Se definido como <code>true</code> , retorna apenas as regras LLD habilitadas que pertencem aos hosts monitorados.
templated	boolean	Se definido como <code>true</code> retornar apenas regras LLD que pertencem a templates.
templateids	string/array	Retorne apenas as regras LLD que pertencem aos templates fornecidos.
selectFilter	query	Retorna uma propriedade <code>filter</code> com dados do filtro usado pela regra LLD.
selectGraphs	query	Retorna uma propriedade <code>graphs</code> com protótipos de gráfico que pertencem à regra LLD.
selectHostPrototypes	query	Suporta <code>count</code> . Retorna uma propriedade <code>hostPrototypes</code> com protótipos de host que pertencem à regra LLD.
selectHosts	query	Suporta <code>count</code> . Retorna uma propriedade <code>hosts</code> com uma array de hosts aos quais a regra LLD pertence.
selectItems	query	Retorne uma propriedade <code>items</code> com protótipos de itens que pertencem à regra LLD.
selectTriggers	query	Suporta <code>count</code> . Retorne uma propriedade <code>triggers</code> com protótipos de gatilho que pertencem à regra LLD.
selectLLDMacroPaths	query	Suporta <code>count</code> . Retorne uma propriedade <code>lld_macro_paths</code> com uma lista de macros LLD e caminhos para valores atribuídos a cada macro correspondente

Parâmetro	Tipo	Descrição
selectPreprocessing	query	<p>Retorne uma propriedade <code>preprocessing</code> propriedade com opções de pré-processamento de regra LLD.</p> <p>Possui as seguintes propriedades:</p> <p><code>type</code> - (string) O tipo de opção de pré-processamento:</p> <ul style="list-style-type: none"> 5 - Correspondência de expressão regular; 11 - XML XPath; 12 - JSONPath; 15 - Não corresponde à expressão regular; 16 - Verificar se há erro em JSON; 17 - Verificar se há erro em XML; 20 - Descarte inalterado com heartbeat; 23 - Prometheus para JSON; 24 - CSV para JSON; 25 - Substituir; 27 - XML para JSON. <p><code>params</code> - (string) Parâmetros adicionais usados pela opção de pré-processamento. Vários parâmetros são separados pelo caractere LF (\n).</p> <p><code>error_handler</code> - (string) Tipo de ação usado em caso de falha na etapa de pré-processamento:</p> <ul style="list-style-type: none"> 0 - A mensagem de erro é definida por Zabbix server; 1 - A mensagem de erro é definida por; 2 - Definir valor personalizado; 3 - Definir mensagem de erro personalizada. <p><code>error_handler_params</code> - (string) Parâmetros do manipulador de erros.</p>
selectOverrides	query	Retorna uma propriedade <code>lld_rule_overrides</code> com uma lista de filtros de substituição, condições e operações que são executadas em objetos protótipo.
filter	object	<p>Retorne apenas os resultados que correspondem exatamente ao filtro fornecido.</p> <p>Aceita uma array, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para corresponder.</p> <p>Suporta filtros adicionais:</p> <p><code>host</code> - nome técnico do host ao qual a regra LLD pertence.</p> <p>Limita o número de registros retornados por subseleções.</p>
limitSelects	integer	<p>Aplica-se às seguintes subseleções:</p> <ul style="list-style-type: none"> <code>selectItems</code>; <code>selectGraphs</code>; <code>selectTriggers</code>. <p>Classifique o resultado pelas propriedades fornecidas.</p>
sortfield	string/array	<p>Os valores possíveis são: <code>itemid</code>, <code>name</code>, <code>key_</code>, <code>delay</code>, <code>type</code> e <code>status</code>.</p> <p>Esses parâmetros sendo comuns para todos os métodos <code>get</code> são descritos em detalhes no reference commentary.</p>
countOutput	boolean	
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	

Parâmetro	Tipo	Descrição
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma série de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperando regras de descoberta de um host

Recupere todas as regras de descoberta para ID de host específico.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.get",
  "params": {
    "output": "extend",
    "hostids": "10202"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "27425",
      "type": "0",
      "snmp_oid": "",
      "hostid": "10202",
      "name": "Network interface discovery",
      "key_": "net.if.discovery",
      "delay": "1h",
      "status": "0",
      "trapper_hosts": "",
      "templateid": "22444",
      "valuemapid": "0",
      "params": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "interfaceid": "119",
      "description": "Discovery of network interfaces as defined in global regular expression \"Netw",
      "lifetime": "30d",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "timeout": "3s",
      "url": "",
      "query_fields": [],
      "posts": "",
      "status_codes": "200",
      "follow_redirects": "1",
    }
  ]
}
```

```

    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "uuid": "",
    "state": "0",
    "error": "",
    "parameters": []
},
{
    "itemid": "27426",
    "type": "0",
    "snmp_oid": "",
    "hostid": "10202",
    "name": "Mounted filesystem discovery",
    "key_": "vfs.fs.discovery",
    "delay": "1h",
    "status": "0",
    "trapper_hosts": "",
    "templateid": "22450",
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "interfaceid": "119",
    "description": "Discovery of file systems of different types as defined in global regular expressions",
    "lifetime": "30d",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "timeout": "3s",
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "uuid": "",
    "state": "0",
    "error": "",
    "parameters": []
}

```

```
],
  "id": 1
}
```

Recuperando condições de filtro

Recupere o nome da regra LLD "24681" e suas condições de filtro. o filtro usa o tipo de avaliação "and", então a propriedade formula é vazio e eval_formula é gerado automaticamente.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.get",
  "params": {
    "output": ["name"],
    "selectFilter": "extend",
    "itemids": ["24681"]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "24681",
      "name": "Filtered LLD rule",
      "filter": {
        "evaltype": "1",
        "formula": "",
        "conditions": [
          {
            "macro": "{#MACRO1}",
            "value": "@regex1",
            "operator": "8",
            "formulaid": "A"
          },
          {
            "macro": "{#MACRO2}",
            "value": "@regex2",
            "operator": "9",
            "formulaid": "B"
          },
          {
            "macro": "{#MACRO3}",
            "value": "",
            "operator": "12",
            "formulaid": "C"
          },
          {
            "macro": "{#MACRO4}",
            "value": "",
            "operator": "13",
            "formulaid": "D"
          }
        ],
        "eval_formula": "A and B and C and D"
      }
    ]
  },
  "id": 1
}
```



```
}
```

Recuperar regra LLD por URL

Recupere a regra LLD para o host pelo valor do campo de URL da regra. Apenas a correspondência exata da string de URL definida para a regra LLD é compatível.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.get",
  "params": {
    "hostids": "10257",
    "filter": {
      "type": 19,
      "url": "http://127.0.0.1/discoverer.php"
    }
  },
  "id": 39,
  "auth": "d678e0b85688ce578ff061bd29a20d3b"
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "28336",
      "type": "19",
      "snmp_oid": "",
      "hostid": "10257",
      "name": "API HTTP agent",
      "key_": "api_discovery_rule",
      "delay": "5s",
      "status": "0",
      "trapper_hosts": "",
      "templateid": "0",
      "valuemapid": "0",
      "params": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "interfaceid": "5",
      "description": "",
      "lifetime": "30d",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "timeout": "3s",
      "url": "http://127.0.0.1/discoverer.php",
      "query_fields": [
        {
          "mode": "json"
        },
        {
          "elements": "2"
        }
      ],
      "posts": "",
      "status_codes": "200",
      "follow_redirects": "1",
    }
  ]
}
```

```

        "post_type": "0",
        "http_proxy": "",
        "headers": {
            "X-Type": "api",
            "Authorization": "Bearer mF_A.B5f-2.1JcM"
        },
        "retrieve_mode": "0",
        "request_method": "1",
        "ssl_cert_file": "",
        "ssl_key_file": "",
        "ssl_key_password": "",
        "verify_peer": "0",
        "verify_host": "0",
        "allow_traps": "0",
        "uuid": "",
        "state": "0",
        "error": "",
        "parameters": []
    },
    ],
    "id": 39
}

```

Recuperar regra LLD com substituições

Recuperar uma regra LLD que tenha várias configurações de substituição.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "discoveryrule.get",
    "params": {
        "output": ["name"],
        "itemids": "30980",
        "selectOverrides": ["name", "step", "stop", "filter", "operations"]
    },
    "id": 39,
    "auth": "d678e0b85688ce578ff061bd29a20d3b"
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": [
        {
            "name": "Discover database host",
            "overrides": [
                {
                    "name": "Discover MySQL host",
                    "step": "1",
                    "stop": "1",
                    "filter": {
                        "evaltype": "2",
                        "formula": "",
                        "conditions": [
                            {
                                "macro": "{#UNIT.NAME}",
                                "operator": "8",
                                "value": "~mysqld\\.service$",
                                "formulaid": "A"
                            }
                        ],
                    },
                    {
                        "macro": "{#UNIT.NAME}",

```

```

        "operator": "8",
        "value": "~mariadb\\.service$",
        "formulaid": "B"
    }
],
    "eval_formula": "A or B"
},
"operations": [
    {
        "operationobject": "3",
        "operator": "2",
        "value": "Database host",
        "opstatus": {
            "status": "0"
        },
        "optag": [
            {
                "tag": "Database",
                "value": "MySQL"
            }
        ],
        "optemplate": [
            {
                "templateid": "10170"
            }
        ]
    }
]
},
{
    "name": "Discover PostgreSQL host",
    "step": "2",
    "stop": "1",
    "filter": {
        "evaltype": "0",
        "formula": "",
        "conditions": [
            {
                "macro": "#{UNIT.NAME}",
                "operator": "8",
                "value": "~postgresql\\.service$",
                "formulaid": "A"
            }
        ],
        "eval_formula": "A"
    },
    "operations": [
        {
            "operationobject": "3",
            "operator": "2",
            "value": "Database host",
            "opstatus": {
                "status": "0"
            },
            "optag": [
                {
                    "tag": "Database",
                    "value": "PostgreSQL"
                }
            ],
            "optemplate": [
                {

```

```

    "templateid": "10263"
  }
]
}
],
"id": 39
}

```

Veja também

- [Graph prototype](#)
- [Host](#)
- [Item prototype](#)
- [LLD rule filter](#)
- [Trigger prototype](#)

Fonte

CDDiscoveryRule::get() in ui/include/classes/api/services/CDDiscoveryRule.php.

discoveryrule.update

Descrição

object discoveryrule.update(object/array lldRules)

Este método permite atualizar as regras LLD existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object/array) Propriedades da regra LLD a serem atualizadas.

A propriedade `itemid` deve ser definida para cada regra LLD, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades da regra LLD padrão](#), o método aceita os seguintes parâmetros.

Parâmetros	Tipo	Descrição
filter	object	LLD regra filter objeto para substituir o filtro atual.
preprocessing	array	Regra LLD preprocessing opções para substituir as opções de pré-processamento atuais.
lld_macro_paths	array	Opções da regra LLD lld_macro_path .
overrides	array	Opções da regra LLD overrides .

Valores de retorno

(object) Retorna um objeto contendo os IDs das regras LLD atualizadas na propriedade `itemids`.

Exemplos

Adicionando um filtro a uma regra LLD

Adicione um filtro para que o conteúdo da macro `{#FSTYPE}` corresponda ao regexp `@File systems for discovery`.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "discoveryrule.update",

```

```

    "params": {
      "itemid": "22450",
      "filter": {
        "evaltype": 1,
        "conditions": [
          {
            "macro": "#{FSTYPE}",
            "value": "@File systems for discovery"
          }
        ]
      }
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
  }
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "22450"
    ]
  },
  "id": 1
}

```

Adicionando caminhos de macro LLD

Request:

```

{
  "jsonrpc": "2.0",
  "method": "discoveryrule.update",
  "params": {
    "itemid": "22450",
    "lld_macro_paths": [
      {
        "lld_macro": "#{MACRO1}",
        "path": "$.json.path"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "22450"
    ]
  },
  "id": 1
}

```

Desativar trapping

Desativar regra de descoberta para LLD trapping.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.update",
  "params": {
    "itemid": "28336",
    "allow_traps": 0
  },
  "id": 36,
  "auth": "d678e0b85688ce578ff061bd29a20d3b"
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "28336"
    ]
  },
  "id": 36
}
```

Atualizando opções de pré-processamento de regra LLD

Atualizar uma regra LLD com a regra de pré-processamento "JSONPath".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.update",
  "params": {
    "itemid": "44211",
    "preprocessing": [
      {
        "type": 12,
        "params": "$.path.to.json",
        "error_handler": 2,
        "error_handler_params": "5"
      }
    ]
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "44211"
    ]
  },
  "id": 1
}
```

Atualizando script de regra LLD

Atualize um script de regra LLD com um script diferente e remova parâmetros desnecessários que foram usados pelo script anterior.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "discoveryrule.update",
```

```

    "params": {
      "itemid": "23865",
      "parameters": [],
      "script": "Zabbix.Log(3, 'Log test');\nreturn 1;"
    },
    "auth": "700ca65537074ec963db7efabda78259",
    "id": 1
  }

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "itemids": [
      "23865"
    ]
  },
  "id": 1
}

```

Fonte

CDiscoveryRule::update() in ui/include/classes/api/services/CDiscoveryRule.php.

Relatório

Esta classe foi projetada para trabalhar com relatórios agendados.

Referências de objetos:

- [Relatório](#)
- [Usuários](#)
- [Grupos de usuários](#)

Métodos disponíveis:

- [report.create](#) - cria novos relatórios programados
- [report.delete](#) - exclui relatórios programados
- [report.get](#) - recupera relatórios programados
- [report.update](#) - atualiza relatórios programados

> objeto de relatório

Os objetos a seguir estão diretamente relacionados à API `report`.

Relatório

O objeto de relatório tem as seguintes propriedades:

Propriedade	Tipo	Descrição
<code>reportid</code>	string	(somente leitura) ID do relatório.
<code>userid</code> (requerido)	string	ID do usuário que criou o relatório.
<code>name</code> (requerido)	string	Nome exclusivo do relatório.
<code>dashboardid</code> (requerido)	string	ID do painel no qual o relatório se baseia.

Propriedade	Tipo	Descrição
period	integer	Período para o qual o relatório será elaborado. Valores possíveis: 0 - (padrão) dia anterior; 1 - semana anterior; 2 - mês anterior; 3 - ano anterior.
start_time	integer	Hora do dia, em segundos, em que o relatório será preparado para envio.
weekdays	integer	Padrão: 0. Dias da semana para envio do relatório. Requerido apenas para relatórios semanais. Os dias da semana são armazenados em formato binário com cada bit representando o dia da semana correspondente. Por exemplo, 12 é igual a 1100 em binário e significa que os relatórios serão enviados todas as quartas e quintas-feiras.
active_since	string	Padrão: 0. Em qual data começar. Valores possíveis: string vazia - (padrão) não especificada (armazenada como 0); data específica no formato AAAA-MM-DD (armazenada como timestamp do início de um dia (00:00:00)).
active_till	string	Em qual data terminar. Valores possíveis: string vazia - (padrão) não especificada (armazenada como 0); data específica no formato AAAA-MM-DD (armazenada como carimbo de hora do fim de um dia (23:59:59)).
subject	string	Reportar assunto da mensagem.
message	string	Texto da mensagem de relatório.
status	integer	Se o relatório está ativado ou desativado. Valores possíveis: 0 - Desativado; 1 - (padrão) Ativado.
description	text	Descrição do relatório.
state	integer	(somente leitura) Estado do relatório. Valores possíveis: 0 - (padrão) o relatório ainda não foi processado; 1 - o relatório foi gerado e enviado com sucesso para todos os destinatários; 2 - falha na geração do relatório; "info" contém informações de erro; 3 - o relatório foi gerado, mas o envio para alguns (ou todos) destinatários falhou; "info" contém informações de erro.
lastsent	timestamp	(somente leitura) Timestamp Unix do último relatório enviado com sucesso.
info	string	(somente leitura) Descrição do erro ou informações adicionais.

Usuários

O objeto de usuários tem as seguintes propriedades:

Propriedade	Tipo	Descrição
userid (requerido)	string	ID do usuário em nome do qual o relatório será gerado.
exclude	integer	0 - (padrão) Gerar relatório por destinatário. Se o usuário deve ser excluído da lista de e-mails. Valores possíveis: 0 - (padrão) Incluir; 1 - Excluir.

Grupos de usuários

O objeto de grupos de usuários tem as seguintes propriedades:

Propriedade	Tipo	Descrição
usrgrpId (requerido)	string	ID do grupo de usuários para o qual enviar o relatório.
access_userid	string	ID do usuário em nome do qual o relatório será gerado. 0 - (padrão) Gerar relatório por destinatário.

report.create

Descrição

`object report.create(object/array reports)`

Este método permite criar novos relatórios agendados.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Relatórios programados para criar.

Além das [propriedades de relatório agendado padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
users	object/array de objetos	Usuários para quem enviar o relatório.
user_groups	object/array de objetos	Grupos de usuários para os quais enviar o relatório.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos relatórios programados criados na propriedade `reportids`. A ordem dos IDs retornados corresponde à ordem dos relatórios programados passados.

Exemplos

Criando um relatório programado

Crie um relatório semanal que será preparado para a semana anterior de segunda a sexta-feira às 12:00 de 01-04-2021 a 31-08-2021.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "report.create",
  "params": {
    "userid": "1",
    "name": "Weekly report",
```

```

    "dashboardid": "1",
    "period": "1",
    "cycle": "1",
    "start_time": "43200",
    "weekdays": "31",
    "active_since": "2021-04-01",
    "active_till": "2021-08-31",
    "subject": "Weekly report",
    "message": "Report accompanying text",
    "status": "1",
    "description": "Report description",
    "users": [
        {
            "userid": "1",
            "access_userid": "1",
            "exclude": "0"
        },
        {
            "userid": "2",
            "access_userid": "0",
            "exclude": "1"
        }
    ],
    "user_groups": [
        {
            "usrgrpid": "7",
            "access_userid": "0"
        }
    ]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "reportids": [
            "1"
        ]
    },
    "id": 1
}

```

Veja também

- [Usuários](#)
- [Grupos de usuários](#)

Fonte

CReport::create() in ui/include/classes/api/services/CReport.php.

report.delete

Descrição

object report.delete(array reportids)

Este método permite excluir relatórios agendados.

Note:

Este método está disponível apenas para o tipo de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos relatórios programados a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos relatórios programados excluídos na propriedade `reportids`.

Exemplos

Excluindo vários relatórios programados

Exclua dois relatórios programados.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "report.delete",
  "params": [
    "1",
    "2"
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "reportids": [
      "1",
      "2"
    ]
  },
  "id": 1
}
```

Fonte

CReport::delete() in ui/include/classes/api/services/CReport.php.

report.get**Descrição**

integer/array `report.get(object parameters)`

O método permite recuperar relatórios programados de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
reportids	string/array	Retornar apenas relatórios programados com os IDs de relatório fornecidos.
expired	boolean	Se definido como true, retorna apenas relatórios programados expirados, se false - somente relatórios programados ativos.
selectUsers	query	Retorne uma propriedade users para a qual o relatório está configurado para ser enviado.
selectUserGroups	query	Retorne uma propriedade user_groups para a qual o relatório está configurado para ser enviado.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: reportid, name, status. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página comentário de referência .
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma array de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando dados do relatório

Request:

```
{
  "jsonrpc": "2.0",
  "method": "report.get",
  "params": [
    "output": "extend",
    "selectUsers": "extend",
    "selectUserGroups": "extend",
    "reportids": ["1", "2"]
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "reportid": "1",
      "userid": "1",
      "name": "Weekly report",
      "dashboardid": "1",
      "period": "1",
      "cycle": "1",
      "start_time": "43200",
    }
  ]
}
```

```

    "weekdays": "31",
    "active_since": "2021-04-01",
    "active_till": "2021-08-31",
    "subject": "Weekly report",
    "message": "Report accompanying text",
    "status": "1",
    "description": "Report description",
    "state": "1",
    "lastsent": "1613563219",
    "info": "",
    "users": [
      {
        "userid": "1",
        "access_userid": "1",
        "exclude": "0"
      },
      {
        "userid": "2",
        "access_userid": "0",
        "exclude": "1"
      }
    ],
    "user_groups": [
      {
        "usrgrpid": "7",
        "access_userid": "0"
      }
    ]
  },
  {
    "reportid": "2",
    "userid": "1",
    "name": "Monthly report",
    "dashboardid": "2",
    "period": "2",
    "cycle": "2",
    "start_time": "0",
    "weekdays": "0",
    "active_since": "2021-05-01",
    "active_till": "",
    "subject": "Monthly report",
    "message": "Report accompanying text",
    "status": "1",
    "description": "",
    "state": "0",
    "lastsent": "0",
    "info": "",
    "users": [
      {
        "userid": "1",
        "access_userid": "1",
        "exclude": "0"
      }
    ],
    "user_groups": []
  }
],
"id": 1
}

```

Veja também

- [Usuários](#)

- [Grupos de usuários](#)

Fonte

CReport::get() in ui/include/classes/api/services/CReport.php.

report.update

Descrição

`object report.update(object/array reports)`

Este método permite atualizar os relatórios agendados existentes.

Note:

Este método está disponível apenas para o tipo de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do relatório agendado a serem atualizadas.

A propriedade `reportid` deve ser definida para cada relatório programado, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de relatório agendado padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
users	object/array of objects	Usuários para substituir os usuários atuais atribuídos ao relatório programado.
user_groups	object/array of objects	Grupos de usuários para substituir os grupos de usuários atuais atribuídos ao relatório programado.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos relatórios programados atualizados na propriedade `reportids`.

Exemplos

Desativando o relatório programado

Request:

```
{
  "jsonrpc": "2.0",
  "method": "report.update",
  "params": {
    "reportid": "1",
    "status": "0"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "reportids": [
      "1"
    ]
  },
  "id": 1
}
```

Veja também

- [Usuários](#)
- [Grupos de usuários](#)

Fonte

CReport::update() in ui/include/classes/api/services/CReport.php.

Script

Esta classe foi projetada para trabalhar com scripts.

Referências de objetos:

- [Script](#)
- [Parâmetros do webhook](#)
- [Debug](#)
- [Entrada de log](#)

Available methods:

- [script.create](#) - criar novos scripts
- [script.delete](#) - excluir scripts
- [script.execute](#) - executar scripts
- [script.get](#) - recuperar scripts
- [script.getscriptsbyhosts](#) - recuperar scripts para hosts
- [script.update](#) - scripts de atualização

> Objeto de script

Os objetos a seguir estão diretamente relacionados à API script.

Script

The script object has the following properties.

Property	Type	Description
scriptid	string	(readonly) ID of the script.
name (required)	string	Name of the script.
type (required)	integer	Script type. Possible values: 0 - Script; 1 - IPMI; 2 - SSH; 3 - Telnet; 5 - (default) Webhook.
command (required)	string	Command to run.
scope	integer	Script scope. Possible values: 1 - default action operation; 2 - manual host action; 4 - manual event action.
execute_on	integer	Where to run the script. Used if type is 0 (script). Possible values: 0 - run on Zabbix agent; 1 - run on Zabbix server; 2 - (default) run on Zabbix server (proxy).

Property	Type	Description
menu_path	string	Folders separated by slash that form a menu like navigation in frontend when clicked on host or event.
authtype	integer	Used if scope is 2 or 4. Authentication method used for SSH script type. Used if type is 2. Possible values: 0 - password; 1 - public key.
username	string	User name used for authentication.
password	string	Required if type is 2 or 3. Password used for SSH scripts with password authentication and Telnet scripts.
publickey	string	Used if type is 2 and authtype is 0 or type is 3. Name of the public key file used for SSH scripts with public key authentication.
privatekey	string	Required if type is 2 and authtype is 1. Name of the private key file used for SSH scripts with public key authentication.
port	string	Required if type is 2 and authtype is 1. Port number used for SSH and Telnet scripts.
groupid	string	Used if type is 2 or 3. ID of the host group that the script can be run on. If set to 0, the script will be available on all host groups.
usrgrpId	string	Default: 0. ID of the user group that will be allowed to run the script. If set to 0, the script will be available for all user groups. Used if scope is 2 or 4.
host_access	integer	Default: 0. Host permissions needed to run the script. Used if scope is 2 or 4. Possible values: 2 - (default) read; 3 - write.

O objeto script tem as seguintes propriedades.

Propriedade	Tipo	Descrição
scriptid	string	(somente leitura) ID do script.
name (requerido)	string	Nome do script.
type (requerido)	integer	Tipo de script. Valores possíveis: 0 - Script; 1 - IPMI; 2 - SSH; 3 - Telnet; 5 - (padrão) Webhook.
command (requerido)	string	Comando para executar.
scope	integer	Escopo do script. Valores possíveis: 1 - operação de ação padrão; 2 - ação de host manual; 4 - ação de evento manual.

Propriedade	Tipo	Descrição
execute_on	integer	Onde executar o script. Usado se type for 0 (script).
menu_path	string	Valores possíveis: 0 - executar no agente Zabbix; 1 - executar no servidor Zabbix ; 2 - (padrão) roda no servidor Zabbix (proxy). Pastas separadas por barra que formam um menu como navegação no frontend quando clicado no host ou evento.
authtype	integer	Usado se scope for 2 ou 4. Método de autenticação usado para o tipo de script SSH. Usado se type for 2.
username	string	Valores possíveis: 0 - senha; 1 - chave pública. Nome de usuário usado para autenticação.
password	string	Obrigatório se type for 2 ou 3. Senha usada para scripts SSH com autenticação de senha e scripts Telnet.
publickey	string	Usado se type for 2 e authtype for 0 ou type for 3. Nome do arquivo de chave pública usado para scripts SSH com autenticação de chave pública.
privatekey	string	Obrigatório se type for 2 e authtype for 1. Nome do arquivo de chave privada usado para scripts SSH com autenticação de chave pública.
port	string	Obrigatório se type for 2 e authtype for 1. Número da porta usado para scripts SSH e Telnet.
groupid	string	Usado se o tipo for 2 ou 3. ID do grupo de hosts no qual o script pode ser executado. Se definido como 0, o script estará disponível em todos os grupos de hosts.
usrgrpuid	string	Padrão: 0. ID do grupo de usuários que terá permissão para executar o script. Se definido como 0, o script estará disponível para todos os grupos de usuários. Usado se scope for 2 ou 4.
host_access	integer	Padrão: 0. Permissões de host necessárias para executar o script. Usado se scope for 2 ou 4.
confirmation	string	Valores possíveis: 2 - (padrão) Leitura; 3 - Escrita. Texto pop-up de confirmação. O pop-up aparecerá ao tentar executar o script do frontend do Zabbix.
timeout	string	Usado se scope for 2 ou 4. Tempo limite de execução do script do webhook em segundos. Sufixos de tempo são suportados, por exemplo 30s, 1m. Obrigatório se type for 5.
parameters	array	Valores possíveis: 1-60s Valor padrão: 30s Array de parâmetros de entrada do webhook .
description	string	Usado se type for 5. Descrição do script.

Propriedade	Tipo	Descrição
confirmation	string	Confirmation pop up text. The pop up will appear when trying to run the script from the Zabbix frontend.
timeout	string	Used if scope is 2 or 4. Webhook script execution timeout in seconds. Time suffixes are supported, e.g. 30s, 1m. Required if type is 5. Possible values: 1-60s Default value: 30s
parameters	array	Array of webhook input parameters . Used if type is 5.
description	string	Description of the script.

Parâmetros do webhook

Os parâmetros passados para o script de webhook quando ele é chamado têm as seguintes propriedades.

Propriedade	Tipo	Descrição
name (requerido)	string	Nome do parâmetro.
value	string	Valor do parâmetro. Suporta macros .

Debug

Informações de depuração do script de webhook executado. O objeto de depuração tem as seguintes propriedades.

Propriedade	Tipo	Descrição
logs	array	Array de entradas de log .
ms	string	Duração da execução do script em milissegundos.

Entrada de log

O objeto de entrada de log tem as seguintes propriedades.

Propriedade	Tipo	Descrição
level	integer	Nível de Log.
ms	string	O tempo decorrido em milissegundos desde que o script foi executado antes que a entrada de log fosse adicionada.
message	string	Mensagem de log.

script.create

Descrição

`script de objeto.create(scripts de objeto/array)`

Este método permite criar novos scripts.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte **Funções do usuário** para obter mais informações.

Parâmetros

(objeto/array) Scripts para criar.

O método aceita scripts com as **propriedades de script padrão**.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos scripts criados na propriedade `scriptids`. A ordem dos IDs retornados corresponde à ordem dos scripts passados.

Exemplos

Criar um script de webhook

Crie um script de webhook que envie uma solicitação HTTP para um serviço externo.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "script.create",
  "params": {
    "name": "Webhook script",
    "command": "try {\n var request = new HttpRequest(),\n response,\n data;\n\n request.addHeader('Co",
    "type": 5,
    "timeout": "40s",
    "parameters": [
      {
        "name": "token",
        "value": "${WEBHOOK.TOKEN}"
      },
      {
        "name": "host",
        "value": "${HOST.HOST}"
      },
      {
        "name": "v",
        "value": "2.2"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "scriptids": [
      "3"
    ]
  },
  "id": 1
}
```

Criar um script SSH

Crie um script SSH com autenticação de chave pública que possa ser executado em um host e tenha um menu de contexto.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "script.create",
  "params": {
    "name": "SSH script",
    "command": "my script command",
    "type": 2,
    "username": "John",
    "publickey": "pub.key",

```

```

        "privatekey": "priv.key",
        "password": "secret",
        "port": "12345",
        "scope": 2,
        "menu_path": "All scripts/SSH",
        "usrgrpuid": "7",
        "groupid": "4"
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "5"
        ]
    },
    "id": 1
}

```

Crie um script personalizado

Crie um script personalizado que reinicializará um servidor. O script exigirá acesso de gravação ao host e exibirá uma mensagem de configuração antes de ser executado no frontend.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "script.create",
    "params": {
        "name": "Reboot server",
        "command": "reboot server 1",
        "confirmation": "Are you sure you would like to reboot the server?",
        "scope": 2,
        "type": 0
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "scriptids": [
            "4"
        ]
    },
    "id": 1
}

```

Fonte

CScript::create() in ui/include/classes/api/services/CScript.php.

script.delete

Descrição

script de objeto.delete(array scriptIds)

Este método permite excluir scripts.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos scripts a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos scripts excluídos na propriedade `scriptids`.

Exemplos

Delete multiple scripts

Delete two scripts.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "script.delete",
  "params": [
    "3",
    "4"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "scriptids": [#### Excluir vários scripts
```

Excluir dois scripts.

Request:

```
```json
{
 "jsonrpc": "2.0",
 "method": "script.delete",
 "params": [
 "3",
 "4"
],
 "auth": "3a57200802b24cda67c4e4010b50c065",
 "id": 1
}
```

Response:

```
{
 "jsonrpc": "2.0",
 "result": {
 "scriptids": [
 "3",
 "4"
]
 },
}
```

```
 "id": 1
}
```

```
 "3",
 "4"
]
```

```
},
"id": 1
}
```

##### Fonte

CScript::delete() in \*ui/include/classes/api/services/CScript.php\*.

##### script.execute {#manual-api-reference-script-execute}  
\

##### Descrição

`script de objeto.execute(parâmetros de objeto)`

Este método permite executar um script em um host ou evento.

::: noteclassic

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser r

:::

##### Parâmetros

`(object)` Parâmetros contendo o ID do script a ser executado e o ID do host ou o ID do evento.

|Parâmetro| [Tipo] (/manual/api/reference\_commentary#data\_types) |Descrição|

|--|--|-----|

|\*\*scriptid\*\*<br>(requerido)|string|ID do script a ser executado.|

|hostid|string|ID do host para executar o script.|

|eventid|string|ID do evento para executar o script.|

##### Valores de retorno

`(object)` Retorna o resultado da execução do script.

|Propriedade| [Tipo] (/manual/api/reference\_commentary#data\_types) |Descrição|

|--|--|-----|

|response|string|Se o script foi executado com sucesso.<br><br>Valores possíveis: `success` ou `failed`.|

|value|string|Saída do script.|

|debug|object|Contém um [objeto de depuração] (/manual/api/reference/script/object#Debug) se um script de w

##### Exemplos

##### Executar um script de webhook

Execute um script de webhook que envia solicitação HTTP para serviço externo.

Request:

```

```json
{
  "jsonrpc": "2.0",
  "method": "script.execute",
  "params": {
    "scriptid": "4",
    "hostid": "30079"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "response": "success",
    "value": "{\"status\":\"sent\",\"timestamp\":\"1611235391\"}",
    "debug": {
      "logs": [
        {
          "level": 3,
          "ms": 480,
          "message": "[Webhook Script] HTTP status: 200."
        }
      ],
      "ms": 495
    }
  },
  "id": 1
}

```

Execute um script personalizado

Execute um script "ping" em um host.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "script.execute",
  "params": {
    "scriptid": "1",
    "hostid": "30079"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "response": "success",
    "value": "PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.\n64 bytes from 127.0.0.1: icmp_req=1 tt",
    "debug": []
  },
  "id": 1
}

```

Fonte

CScript::execute() in ui/include/classes/api/services/CScript.php.

script.get

Descrição

`integer/array script.get(object parameters)`

O método permite recuperar scripts de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
groupids	string/array	Retorna apenas scripts que podem ser executados nos grupos de hosts fornecidos.
hostids	string/array	Retorna apenas scripts que podem ser executados nos hosts fornecidos.
scriptids	string/array	Retornar apenas scripts com os IDs fornecidos.
usrgrpid	string/array	Retorna apenas scripts que podem ser executados por usuários em determinados grupos de usuários.
selectGroups	query	Retorna uma propriedade groups com grupos de hosts nos quais o script pode ser executado.
selectHosts	query	Retorna uma propriedade hosts com hosts nos quais o script pode ser executado.
selectActions	query	Retorna uma propriedade actions com ações às quais o script está associado.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: scriptid e name . Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma array de objetos;
- a contagem de objetos recuperados, se o parâmetro **countOutput** foi usado.

Exemplos

Recuperar todos os scripts

Recupere todos os scripts configurados.

Request:


```
{
  "jsonrpc": "2.0",
  "method": "script.get",
  "params": {
    "output": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "scriptid": "1",
      "name": "Ping",
      "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
      "host_access": "2",
      "usrgrpid": "0",
      "groupid": "0",
      "description": "",
      "confirmation": "",
      "type": "0",
      "execute_on": "1",
      "timeout": "30s",
      "parameters": []
    },
    {
      "scriptid": "2",
      "name": "Traceroute",
      "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
      "host_access": "2",
      "usrgrpid": "0",
      "groupid": "0",
      "description": "",
      "confirmation": "",
      "type": "0",
      "execute_on": "1",
      "timeout": "30s",
      "parameters": []
    },
    {
      "scriptid": "3",
      "name": "Detect operating system",
      "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
      "host_access": "2",
      "usrgrpid": "7",
      "groupid": "0",
      "description": "",
      "confirmation": "",
      "type": "0",
      "execute_on": "1",
      "timeout": "30s",
      "parameters": []
    },
    {
      "scriptid": "4",
      "name": "Webhook",
      "command": "try {\n var request = new HttpRequest(),\n response,\n data;\n\n request.addHeader",
      "host_access": "2",
      "usrgrpid": "7",

```

```

        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "5",
        "execute_on": "1",
        "timeout": "30s",
        "parameters": [
            {
                "name": "token",
                "value": "${WEBHOOK.TOKEN}"
            },
            {
                "name": "host",
                "value": "${HOST.HOST}"
            },
            {
                "name": "v",
                "value": "2.2"
            }
        ]
    },
    "id": 1
}

```

Veja também

- [Host](#)
- [Grupo de hosts](#)

Fonte

CScript::get() in ui/include/classes/api/services/CScript.php.

script.getscriptsbyhosts

Descrição

script de objeto.getscriptsbyhosts(array hostIds)

Este método permite recuperar scripts disponíveis nos hosts fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(string/array) IDs de hosts para retornar para scripts.

Valores de retorno

(object) Retorna um objeto com IDs de host como propriedades e arrays de scripts disponíveis como valores.

Note:

O método expandirá automaticamente as macros no texto de "confirmação".

Exemplos

Recuperar scripts por IDs de host

Recupere todos os scripts disponíveis nos hosts "30079" e "30073".

Request:

```

{
    "jsonrpc": "2.0",

```

```

"method": "script.getscriptsbyhosts",
"params": [
  "30079",
  "30073"
],
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "30079": [
      {
        "scriptid": "3",
        "name": "Detect operating system",
        "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpuid": "7",
        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
      },
      {
        "scriptid": "1",
        "name": "Ping",
        "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpuid": "0",
        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
      },
      {
        "scriptid": "2",
        "name": "Traceroute",
        "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpuid": "0",
        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
      }
    ],
    "30073": [
      {
        "scriptid": "3",
        "name": "Detect operating system",
        "command": "sudo /usr/bin/nmap -O {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpuid": "7",
        "groupid": "0",

```

```

        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
    },
    {
        "scriptid": "1",
        "name": "Ping",
        "command": "/bin/ping -c 3 {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpid": "0",
        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
    },
    {
        "scriptid": "2",
        "name": "Traceroute",
        "command": "/usr/bin/traceroute {HOST.CONN} 2>&1",
        "host_access": "2",
        "usrgrpid": "0",
        "groupid": "0",
        "description": "",
        "confirmation": "",
        "type": "0",
        "execute_on": "1",
        "hostid": "10001"
    }
]
},
"id": 1
}

```

Fonte

CScript::getScriptsByHosts() in ui/include/classes/api/services/CScript.php.

script.update

Descrição

script de objeto.update(scripts de objeto/array)

Este método permite atualizar os scripts existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) **Propriedades do script** a ser atualizado.

A propriedade scriptid deve ser definida para cada script, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas. Uma exceção é a mudança da propriedade type de 5 (Webhook) para outra: a propriedade parameters será limpa.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos scripts atualizados na propriedade scriptids.

Exemplos

Alterar comando de script

Altere o comando do script para `"/bin/ping -c 10 {HOST.CONN} 2>&1"`. Request:

```
{
  "jsonrpc": "2.0",
  "method": "script.update",
  "params": {
    "scriptid": "1",
    "command": "/bin/ping -c 10 {HOST.CONN} 2>&1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "scriptids": [
      "1"
    ]
  },
  "id": 1
}
```

Fonte

CScript::update() in ui/include/classes/api/services/CScript.php.

Serviço

Esta classe foi projetada para trabalhar com infraestrutura de TI/serviços de negócios.

Referências de objetos:

- [Serviço](#)
- [Regra de status](#)
- [Etiqueta de serviço](#)
- [Alarme de serviço](#)
- [Etiqueta do problema](#)

Métodos disponíveis:

- [service.create](#) - creating new services
- [service.delete](#) - deleting services
- [service.get](#) - retrieving services
- [service.update](#) - updating services

> Objeto de serviço

Os objetos a seguir estão diretamente relacionados à API service.

Serviço

O objeto de serviço tem as seguintes propriedades.

Propriedade	Descrição
serviceid	(somente leitura) ID do serviço.

Propriedade	Descrição
algorithm integer (requerido)	Regra de cálculo de status. Aplicável apenas se existirem serviços filhos. Valores possíveis: 0 - definir o status como OK; 1 - mais crítico se todos os serviços filhos tiverem problemas; 2 - mais crítico dos serviços filho.
name string (requerido)	Nome do serviço.
sortorder integer (requerido)	Posição do serviço usado para classificação. Valores possíveis: 0-999.
weight integer	Peso do serviço. Valores possíveis: 0-1000000.
propagation_rule integer	Padrão: 0. Regra de propagação de status. Deve ser definido junto com propagation_value . Valores possíveis: 0 - (padrão) propagar o status do serviço como está - sem alterações; 1 - aumentar o status propagado em um determinado propagation_value (por 1 a 5 gravidades); 2 - diminuir o status propagado por um determinado propagation_value (por 1 a 5 gravidades); 3 - ignorar este serviço - o status não é propagado para o serviço pai; 4 - defina o status do serviço fixo usando um determinado propagation_value .
propagation_value integer	Valor de propagação de status. Deve ser definido junto com propagation_rule . Valores possíveis para propagation_rule com valores 0 e 3: 0. Valores possíveis para propagation_rule com valores 1 e 2: 1-5. Possíveis valores para propagation_rule com valor 4: -1 - OK; 0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Médio; 4 - Alto; 5 - Desastre.
status integer	(somente leitura) Se o serviço está OK ou com problema. Se o serviço estiver com problema, status é igual a: - a gravidade do problema mais crítico; - o status mais alto de um serviço filho em estado de problema. Se o serviço estiver em estado OK, status será igual a -1.
description string	Descrição do serviço.
uuid string	Identificador único universal. Para operações de atualização, este campo é somente leitura.
created integer	Timestamp Unix quando o serviço foi criado.
readonly boolean	(readonly) Acesso ao serviço. Valores possíveis: 0 - leitura-gravação; 1 - somente leitura.

Regra de status

O objeto de regra de status tem as seguintes propriedades.

Propriedade	Tipo	Descrição
type (requerido)	integer	Condição para definir o status (status Novo). Valores possíveis: 0 - se pelo menos (N) serviços filho tiverem status (Status) ou superior; 1 - se pelo menos (N%) dos serviços de apoio à criança têm estatuto (Status) ou superior; 2 - se menos de (N) serviços infantis tiverem status (Status) ou inferior; 3 - se menos de (N%) dos serviços à criança tiverem status (Status) ou inferior; 4 - se o peso dos serviços infantis com status (Status) ou superior for no mínimo (W); 5 - se o peso dos serviços infantis com status (Status) ou superior for pelo menos (N%); 6 - se o peso dos serviços infantis com status (Status) ou inferior for menor que (W); 7 - se o peso dos serviços filho com status (Status) ou abaixo for menor que (N%). Onde: - N (W) é <code>limit_value</code> ; - (Status) é <code>limit_status</code> ; - (status Novo) é <code>new_status</code> .
limit_value (requerido)	integer	Valor limite. Valores possíveis: - para N e W: 1-100000; - para N%: 1-100.
limit_status (requerido)	integer	Status do limite. Valores possíveis: -1 - OK; 0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Média; 4 - Alta; 5 - Desastre.
new_status (requerido)	integer	Novo valor de status. Valores possíveis: 0 - Não classificado; 1 - Informação; 2 - Aviso; 3 - Médio; 4 - Alto; 5 - Desastre.

Etiqueta de serviço

O objeto de etiqueta de serviço tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tag (required)	string	Nome da etiqueta de serviço.
value	string	Valor da etiqueta de serviço.

Alarme de serviço

Note:

Os alarmes de serviço não podem ser criados, atualizados ou excluídos diretamente pela API do Zabbix.

Os objetos de alarme de serviço representam a mudança de estado de um serviço. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
clock	timestamp	Hora em que a mudança de estado do serviço aconteceu.
value	integer	Status do serviço.
		Consulte a propriedade do status do serviço para obter uma lista de valores possíveis.

Etiqueta do problema

As etiquetas de problema permitem vincular serviços a eventos de problema. O objeto de etiquetas de problema tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tag (requerido)	string	Nome da etiqueta do problema.
operator	integer	Operador de condição de mapeamento. Valores possíveis: 0 - (padrão) é igual; 2 - igual.
value	string	Valor da etiqueta do problema.

service.create

Descrição

`object service.create(object/array services)`

Este método permite criar novos serviços.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte **Funções do usuário** para obter mais informações.

Parâmetros

serviços (objeto/array) para criar.

Além das **propriedades de serviço padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
children	array	Serviços filho a serem vinculados ao serviço.
parents	array	Os serviços filhos devem ter a propriedade <code>serviceid</code> definida. Serviços pai a serem vinculados ao serviço.
tags	array	Os serviços pais devem ter a propriedade <code>serviceid</code> definida. Etiqueta de serviço a ser criada para o serviço.
problem_tags	array	Etiquetas de problema a serem criadas para o serviço.
status_rules	array	Regras de status a serem criadas para o serviço.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos serviços criados na propriedade `serviceids`. A ordem dos IDs retornados corresponde à ordem dos serviços passados

Exemplos

Criando um serviço

Crie um serviço que será alterado para o estado de problema, se pelo menos um serviço filho tiver um problema.

Request:


```
{
  "jsonrpc": "2.0",
  "method": "service.create",
  "params": {
    "name": "Server 1",
    "algorithm": 1,
    "sortorder": 1
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "serviceids": [
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CService::create() in ui/include/classes/api/services/CService.php.

service.delete

Descrição

object service.delete(array serviceIds)

Este método permite excluir serviços.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos serviços a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos serviços excluídos na propriedade serviceids.

Exemplos

Excluindo vários serviços

Excluir dois serviços.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.delete",
  "params": [
    "4",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "serviceids": [
      "4",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CService::delete() in ui/include/classes/api/services/CService.php.

service.get

Descrição

integer/array service.get(object parameters)

O método permite recuperar serviços de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro

Parâmetro	Tipo	Descrição
serviceids	string/array	Retornar apenas serviços com os IDs fornecidos.
parentids	string/array	Retornar apenas serviços vinculados aos serviços pai fornecidos.
deep_parentids	flag	Retornar todos os serviços filhos diretos e indiretos. Usado junto com parentids.
childids	string/array	Retornar apenas serviços vinculados aos serviços filho fornecidos.
evaltype	integer	Regras para pesquisa de etiquetas.
Valores possíveis:		
0 - (padrão) E/Ou;		
2 - Ou.		
tags	object/array of objects	Retornar apenas serviços com as etiquetas fornecidas. Correspondência exata por etiqueta e pesquisa com distinção entre maiúsculas e minúsculas por valor de etiqueta, dependendo do valor do operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Uma array vazia retorna todos os serviços.
Valores de operador possíveis:		
0 - (padrão) Contém;		
1 - Igual a;		
2 - Não contém;		
3 - Não é igual;		
4 - Existe;		
5 - Não existe.		

Parâmetro	Tipo	
problemTags	string/array of objects	Retornar apenas serviços com as etiquetas de problema fornecidas. Correspondência exata por etiqueta e pesquisa com distinção entre maiúsculas e minúsculas por valor de etiqueta, dependendo do valor do operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Um array vazio retorna todos os serviços. Possíveis valores de operador: 0 - (padrão) Contém; 1 - É igual; 2 - Não contém; 3 - Não é igual; 4 - Existe; 5 - Não existe.
withoutProblemTags	boolean	Retornar apenas serviços sem etiquetas de problema.
slaids	string/array	Retornar apenas os serviços vinculados ao(s) SLA(s) específico(s).
selectChildren	boolean	Retornar uma propriedade children com os serviços filho. Suporta count.
selectParents	boolean	Retornar uma propriedade parents com os serviços pai. Suporta count.
selectTags	query	Retornar uma propriedade etiquetas com etiquetas de serviço. Suporta count.
selectProblemEvents	boolean	Retornar uma propriedade problem_events com uma matriz de objetos de evento de problema. O objeto de evento de problema tem as seguintes propriedades: eventid - (string) ID do evento; severity - (string) Gravidade do evento atual; name - (string) Nome do evento resolvido. Suporta count.
selectProblemTags	boolean	Retornar uma propriedade etiquetas de problema com etiquetas de problema. Suporta count.
selectStatusRules	boolean	Return a status_rules property with status rules. Suporta count.
selectStatusTimeline	boolean/object/array of objects	Retorne uma propriedade status_timeline contendo alterações de estado do serviço para determinados períodos. Formato [{"period_from": "<period_from>", "period_to": "<period_to>"}, ...] - period_from sendo uma data inicial (inclusive; timestamp inteiro) e period_to sendo uma data final (exclusiva; timestamp inteiro) para o período em que você está interessado. Retorna um array de entradas contendo uma propriedade start_value e um array alarmes para as mudanças de estado dentro de períodos especificados.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas. Os valores possíveis são: serviceid, name, status, sortorder e 'created_at'. countOutput boolean Esses parâmetros sendo comuns para todos os métodosget' são descritos em detalhes no comentário de referência.
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

searchWildcardsEnabled	boolean
sortorder	string/array
startSearch	boolean

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando todos os serviços

Recupere todos os dados sobre todos os serviços e suas dependências.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.get",
  "params": {
    "output": "extend",
    "selectChildren": "extend",
    "selectParents": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "serviceid": "1",
      "name": "My Service - 0001",
      "status": "-1",
      "algorithm": "2",
      "sortorder": "0",
      "weight": "0",
      "propagation_rule": "0",
      "propagation_value": "0",
      "description": "My Service Description 0001.",
      "uuid": "dfa4daeaea754e3a95c04d6029182681",
      "created_at": "946684800",
      "readonly": false,
      "parents": [],
      "children": []
    },
    {
      "serviceid": "2",
      "name": "My Service - 0002",
      "status": "-1",
      "algorithm": "2",
      "sortorder": "0",
      "weight": "0",
      "propagation_rule": "0",
      "propagation_value": "0",
      "description": "My Service Description 0002.",
      "uuid": "20ea0d85212841219130abeaca28c065",
      "created_at": "946684800",
      "readonly": false,

```

```

        "parents": [],
        "children": []
    }
],
    "id": 1
}

```

Fonte

CService::get() in ui/include/classes/api/services/CService.php.

service.update

Descrição

object service.update(object/array services)

Este método permite atualizar os serviços existentes.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parameters

(object/array) propriedades de serviço a serem atualizadas.

A propriedade `serviceid` deve ser definida para cada serviço, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de serviço padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
children	array	Serviços filho para substituir os serviços filhos do serviço atual.
parents	array	Os serviços filhos devem ter a propriedade <code>serviceid</code> definida. Serviços pai para substituir os pais de serviço atuais.
tags	array	Os pais devem ter a propriedade <code>serviceid</code> definida.
problem_tags	array	Etiquetas de serviço para substituir as etiquetas de serviço atuais.
status_rules	array	Etiquetas de problema para substituir as etiquetas de problema atuais.
		Regras de status para substituir as regras de status atuais.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos serviços atualizados na propriedade `serviceids`.

Exemplos

Configurando um serviço pai para um serviço

Faça o serviço com ID "3" ser o pai do serviço com ID "5". Request:

```

{
    "jsonrpc": "2.0",
    "method": "service.update",
    "params": {
        "serviceid": "5",
        "parents": [
            {
                "serviceid": "3"
            }
        ]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
}

```

```
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "serviceids": [
      "5"
    ]
  },
  "id": 1
}
```

Adicionando um tempo de inatividade programado

Adicione um tempo de inatividade para serviço com ID "4" agendado semanalmente de segunda-feira às 22:00 até terça-feira às 10:00. Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.update",
  "params": {
    "serviceid": "4",
    "times": [
      {
        "type": "1",
        "ts_from": "165600",
        "ts_to": "201600"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "serviceids": [
      "4"
    ]
  },
  "id": 1
}
```

Fonte

CService::update() in ui/include/classes/api/services/CService.php.

Serviço de limpeza

Esta classe é projetada para trabalhar com tarefas domésticas.

Referências de objetos:

- **Arrumação**

Métodos disponíveis:

- **housekeeping.get** - recuperar a limpeza
- **housekeeping.update** - atualizar a limpeza

> Housekeeping object

Os objetos a seguir estão diretamente relacionados à API housekeeping.

Serviço de limpeza

O objeto de configurações tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----|-----| |hk_events_mode|integer|Ativar limpeza interna para eventos e alertas.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_events_trigger|string|Período de armazenamento de dados do acionador. Aceita segundos e unidade de tempo com sufixo.

Padrão: 365d.| |hk_events_service|string|Período de armazenamento de dados do serviço. Aceita segundos e unidade de tempo com sufixo.

Padrão: 1d.| |hk_events_internal|string|Período de armazenamento de dados internos. Aceita segundos e unidade de tempo com sufixo.

Padrão: 1d.| |hk_events_discovery|string|Período de armazenamento de dados de descoberta de rede. Aceita segundos e unidade de tempo com sufixo.

Padrão: 1d.| |hk_events_autoreg|string|Período de armazenamento de dados de registro automático. Aceita segundos e unidade de tempo com sufixo.

Padrão: 1d.| |hk_services_mode|integer|Ativar limpeza interna para serviços.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_services|string|Período de armazenamento de dados de serviços. Aceita segundos e unidade de tempo com sufixo.

Padrão: 365d.| |hk_audit_mode|integer|Ativar manutenção interna para auditoria.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_audit|string|Período de armazenamento de dados de auditoria. Aceita segundos e unidade de tempo com sufixo.

Padrão: 365d.| |hk_sessions_mode|integer|Ativar limpeza interna para sessões.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_sessions|string|Período de armazenamento de dados da sessão. Aceita segundos e unidade de tempo com sufixo.

Padrão: 365d.| |hk_history_mode|integer|Ativar manutenção interna para histórico.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_history_global|integer|Substituir o período do histórico do item.

Valores possíveis:
0 - Não substituir;
1 - (padrão) Substituir.| |hk_history|string|Período de armazenamento de dados do histórico. Aceita segundos e unidade de tempo com sufixo.

Padrão: 90d.| |hk_trends_mode|integer|Ativar manutenção interna para tendências.

Valores possíveis:
0 - Desativar;
1 - (padrão) Ativar.| |hk_trends_global|integer|Substituir período de tendência do item.

Valores possíveis:
0 - Não substituir;
1 - (padrão) substituir.| |hk_trends|string|Período de armazenamento de dados do Trends. Aceita segundos e unidade de tempo com sufixo.

Padrão: 365d.| |db_extension|string|(readonly) Extensão do banco de dados do sinalizador de configuração. Se esse sinalizador estiver definido como "timescaledb", o servidor altera seu comportamento para limpeza e exclusão de item.| |compression_status|integer|Ative a compactação do TimescaleDB para histórico e tendências.

Valores possíveis:
0 - (padrão) Desativado;
1 - Ativado.| |compress_older|string|Comprimir registros de histórico e tendências anteriores ao período especificado. Aceita segundos e unidade de tempo com sufixo.

Padrão: 7d.| |compressão_disponibilidade|inteiro|(somente leitura) Disponibilidade de compactação.

Valores possíveis:
0 - Indisponível;
1 - Disponível.|

housekeeping.get

Descrição

`object housekeeping.get(object parameters)`

O método permite recuperar o objeto de limpeza de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta apenas um parâmetro.

[Parâmetro|**Tipo**|Descrição] |-----|-----|-----| |output|query|Este parâmetro é comum para todos os métodos get descritos no [comentário de referência](#).|

Valores de retorno

(object) Retorna o objeto de limpeza.

Exemplos

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "arrumação.get",
  "parâmetros": {
    "saída": "estender"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": {
    "hk_events_mode": "1",
    "hk_events_trigger": "365d",
    "hk_events_service": "1d",
    "hk_events_internal": "1d",
    "hk_events_discovery": "1d",
    "hk_events_autoreg": "1d",
    "hk_services_mode": "1",
    "hk_services": "365d",
    "hk_audit_mode": "1",
    "hk_audit": "365d",
    "hk_sessions_mode": "1",
    "hk_sessions": "365d",
    "hk_history_mode": "1",
    "hk_history_global": "0",
    "hk_history": "90d",
    "hk_trends_mode": "1",
    "hk_trends_global": "0",
    "hk_trends": "365d",
    "db_extension": "",
    "compression_status": "0",
    "compress_older": "7d"
  },
  "id": 1
}
```

Fonte

CHousekeeping ::get() in ui/include/classes/api/services/CHousekeeping.php.

housekeeping.update

Descrição

object housekeeping.update(object housekeeping)

Este método permite atualizar as configurações de limpeza existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [Do utilizador funções](#) Para maiores informações.

Parâmetros

(object) Propriedades de limpeza a serem atualizadas.

Valores de retorno

(array) Retorna o array com os nomes dos parâmetros atualizados.

Exemplos

Solicitação:


```
{
  "jsonrpc": "2.0",
  "método": "arrumação.atualização",
  "parâmetros": {
    "hk_events_mode": "1",
    "hk_events_trigger": "200d",
    "hk_events_internal": "2d",
    "hk_events_discovery": "2d"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    "hk_events_mode",
    "hk_events_trigger",
    "hk_events_internal",
    "hk_events_discovery"
  ],
  "id": 1
}
```

Fonte

CHousekeeping::update() em ui/include/classes/api/services/CHousekeeping.php.

Serviço descoberto

Esta classe foi projetada para funcionar com serviços descobertos.

Referências de objetos:

- [Descoberto serviço](#)

Métodos disponíveis:

- [dservice.get](#) - recuperar serviços descobertos

> Objeto de serviço descoberto

Os objetos a seguir estão diretamente relacionados à API dservice.

Serviço descoberto

Note:

Os serviços descobertos são criados pelo servidor Zabbix e não pode ser modificado por meio da API.

O objeto de serviço descoberto contém informações sobre um serviço descoberto por uma regra de descoberta de rede em um host. Tem o seguinte propriedades.

Propriedade	Tipo	Descrição
dserviceid	string	ID do serviço descoberto.
dcheckid	string	ID da verificação de descoberta usada para detectar o serviço.
dhostid	string	ID do host descoberto executando o serviço.
dns	string	DNS do host que executa o serviço.
ip	string	endereço IP do host que executa o serviço.
lastdown	timestamp	Horário em que o serviço descoberto foi desativado pela última vez.
lastup	timestamp	Horário em que o serviço descoberto foi ativado pela última vez.
port	inteiro	Número da porta do serviço.
status	inteiro	Status do serviço.

Valores possíveis:
0 - serviço ativo;
1 - serviço desativado.
|valor| string|Valor retornado pelo serviço ao realizar uma verificação de descoberta do agente Zabbix, SNMPv1, SNMPv2 ou SNMPv3.

#####dservice.get {#manual-api-reference-dservice-get}

Descrição

integer/array dservice.get(object parameters)

O método permite recuperar serviços descobertos de acordo com o parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Usuário funções](#) Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|Tipo|Descrição] |-----|-----|-----|-----| |dserviceids|string/array|Retorna apenas serviços descobertos com os IDs fornecidos.| |dhostids|string/array|Retorna apenas serviços descobertos que pertencem aos hosts descobertos fornecidos.| |dcheckids|string/array|Retorna apenas os serviços descobertos que foram detectados pelas verificações de descoberta fornecidas.| |druleids|string/array|Retorna apenas os serviços descobertos que foram detectados pelas regras de descoberta fornecidas.| |selectDRules|query|Retorne uma propriedade **drules** com uma matriz das regras de descoberta que detectaram o serviço.| |selectDHosts|query|Retorne uma propriedade **dhosts** com uma matriz dos hosts descobertos aos quais o serviço pertence.| |selectHosts|query|Retorne uma propriedade **hosts** com os hosts com o mesmo endereço IP e proxy do serviço.

Suporta count.| |limitSelects|integer|Limite o número de registros retornados por subseleções.

Aplica-se às seguintes subseleções:
selectHosts - o resultado será classificado por hostid.| |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: dserviceid, dhostid e ip.| |countOutput|boolean|Estes parâmetros sendo comuns para todos os métodos get são descritos em detalhes no **comentário de referência**.| |editável|boolean|^| |excludePesquisa|boolean|^| |filtro|objeto|^| |limite|inteiro|^| |saída|consulta|^| |preservekeys|boolean|^| |pesquisar|objeto|^| |searchByAny|boolean|^| |searchWildcardsEnabled|boolean|^| |ordem de classificação|cadeia/matriz|^| |startSearch|boolean|^|

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput tiver sido usado.

Exemplos

Recuperar serviços descobertos em um host

Recupere todos os serviços descobertos detectados no host descoberto "11".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dservice.get",
  "parâmetros": {
    "saída": "estender",
    "dhostids": "11"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "dserviceid": "12",
      "dhostid": "11",
      "valor": "",
      "porta": "80",
      "estado": "1",
      "último": "0",
      "lastdown": "1348650607",
    }
  ]
}
```

```

        "dcheckid": "5",
        "ip": "192.168.1.134",
        "dns": "john.local"
    },
    {
        "dserviceid": "13",
        "dhostid": "11",
        "valor": "",
        "porta": "21",
        "estado": "1",
        "último": "0",
        "lastdown": "1348650610",
        "dcheckid": "6",
        "ip": "192.168.1.134",
        "dns": "john.local"
    }
],
"id": 1
}

```

Veja também

- [Descoberto host](#)
- [Descoberta check](#)
- [Host](#)

Fonte

CDServic::get() em ui/include/classes/api/services/CDServic.php.

SLA

Esta classe é projetada para trabalhar com objetos SLA (Service Level Agreement) usados para estimar o desempenho de Infraestrutura de TI e serviços de negócios.

Referências de objetos:

- [SLA](#)
- [SLA schedule](#)
- [SLA excluded downtime](#)
- [SLA service tag](#)

Métodos disponíveis:

- [sla.create](#) - criando novos SLAs
- [sla.delete](#) - excluindo SLAs
- [sla.get](#) - recuperando SLAs
- [sla.getsli](#) - recuperando informações de disponibilidade como Indicador de Nível de Serviço (SLI)
- [sla.update](#) - atualizando SLA

> objeto SLA

Os objetos a seguir estão diretamente relacionados à API `sla` (Service Level Agreement).

SLA

O objeto SLA tem as seguintes propriedades.

Propriedade	Tipo	Descrição
slaid	string	(somente leitura) ID do SLA.

Propriedade	Tipo	Descrição
name (requerido)	string	Nome do SLA.
period (requerido)	integer	Período de reporte do SLA. Valores possíveis: 0 - diariamente; 1 - semanalmente; 2 - mensalmente; 3 - trimestralmente; 4 - anualmente.
slo (requerido)	float	Objetivo de nível de serviço mínimo aceitável expresso em porcentagem. Se o Indicador de nível de serviço (SLI) cair, o SLA é considerado com problema/estado não cumprido.
effective_date	integer	Valores possíveis: 0-100 (até 4 dígitos fracionários). Data de vigência do SLA.
timezone (requerido)	string	Valores possíveis: data e hora em UTC. Relatório de fuso horário, por exemplo: Europe/London, UTC. Para obter a lista completa de fusos horários compatíveis, consulte documentação do PHP .
status	integer	Status do SLA. Valores possíveis: 0 - (padrão) SLA desativado; 1 - SLA ativado.
description	string	Descrição do SLA.

Cronograma de SLA

O objeto de agendamento SLA define os períodos em que os serviços conectados são agendados para funcionar. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
period_from (requerido)	integer	Hora de início do período de tempo semanal recorrente (inclusive). Valores possíveis: número de segundos (contando a partir de domingo).
period_to (requerido)	integer	Hora de término do período de tempo semanal recorrente (exclusivo). Valores possíveis: número de segundos (contando a partir de domingo).

Tempo de inatividade excluído do SLA

O objeto de tempo de inatividade excluído define períodos em que o(s) serviço(s) conectado(s) está(ão) programado(s) para não funcionar, sem afetar o SLI, por exemplo, em manutenção planejada.

Propriedade	Tipo	Descrição
name (requerido)	string	Nome do tempo de inatividade excluído.
period_from (requerido)	integer	Hora de início do tempo de inatividade excluído (inclusive). Valores possíveis: timestamp.
period_to (requerido)	integer	Tempo de término do tempo de inatividade excluído (exclusivo). Valores possíveis: timestamp.

Etiqueta de serviço SLA

O objeto da etiqueta de serviço do SLA vincula os serviços a serem incluídos nos cálculos do SLA. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
tag (requerido)	string	Nome da etiqueta de serviço SLA.
operator	integer	Operador de etiqueta de serviço SLA. Valores possíveis: 0 - (padrão) equals; 2 - like
value	string	Valor da etiqueta de serviço SLA.

sla.create

Descrição

object `sla.create(object/array SLAs)`

Este método permite criar novos objetos SLA.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Objetos SLA para criar.

Além das [propriedades de SLA padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
service_tags (requerido)	array	Etiquetas de serviço SLA a serem criadas para o SLA. Pelo menos uma etiqueta de serviço deve ser especificada.
schedule	array	Programação de SLA a ser criada para o SLA. A especificação de um parâmetro vazio será interpretada como uma programação 24 horas por dia, 7 dias por semana. Padrão: programação 24 horas por dia, 7 dias por semana.
excluded_downtime	array	SLA excluídos tempos de inatividade a serem criados para o SLA.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos SLAs criados na propriedade `sla_ids`. A ordem dos IDs retornados corresponde à ordem dos SLAs passados.

Exemplos

Criando um SLA

Instrua para criar uma entrada de SLA para:

* acompanhamento do tempo de atividade para serviços relacionados ao mecanismo SQL; * programação personalizada de todos os dias da semana, excluindo a última hora no sábado;

* uma data efetiva do último dia do ano de 2022;

* com tempo de inatividade planejado de 1 hora e 15 minutos a partir da meia-noite do dia 4 de julho; * O cálculo do relatório semanal do SLA estará ativado; * O SLO mínimo aceitável será de 99,9995%.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "sla.create",
  "params": [
    {
      "name": "Database Uptime",
      "slo": "99.9995",
```

```

        "period": "1",
        "timezone": "America/Toronto",
        "description": "Provide excellent uptime for main database engines.",
        "effective_date": 1672444800,
        "status": 1,
        "schedule": [
            {
                "period_from": 0,
                "period_to": 601200
            }
        ],
        "service_tags": [
            {
                "tag": "Database",
                "operator": "0",
                "value": "MySQL"
            },
            {
                "tag": "Database",
                "operator": "0",
                "value": "PostgreSQL"
            }
        ],
        "excluded_downtimes": [
            {
                "name": "Software version upgrade rollout",
                "period_from": "1648760400",
                "period_to": "1648764900"
            }
        ]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "slays": [
            "5"
        ]
    },
    "id": 1
}

```

Fonte

CSla::create() in ui/include/classes/api/services/CSla.php.

sla.delete

Descrição

object sla.delete(array slays)

Este método permite excluir entradas de SLA.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos SLAs a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos SLAs excluídos na propriedade `slאים`.

Exemplos

Como excluir vários SLAs

Excluir duas entradas de SLA.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "sla.delete",
  "params": [
    "4",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "slאים": [
      "4",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

`CSla::delete()` in `ui/include/classes/api/services/CSla.php`.

#####`sla.get` {#manual-api-reference-sla-get}

Descrição

`integer/array sla.get(object parameters)`

O método permite recuperar objetos SLA de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
<code>slאים</code>	<code>string/array</code>	Retornar apenas SLAs com os IDs fornecidos.
<code>serviceids</code>	<code>string/array</code>	Retornar apenas SLAs que correspondam aos serviços específicos.
<code>selectSchedulequery</code>		Retorne uma propriedade <code>schedule</code> com agendamentos de SLA.
		Suporta <code>count</code> .

Parâmetro	Tipo	Descrição
selectExcludedDowntimes	query	Retorne uma propriedade excluded_downtimes com tempos de inatividade excluídos por SLA.
selectServiceTags	query	Suporta count. Retorne uma propriedade service_tags com etiquetas de serviço SLA.
		Suporta count. sortfield string/array Classifique o resultado pelas propriedades fornecidas.
		Os valores possíveis são:slaid,name,period,slo,effective_date,timezone,statusdescription . countOutput boolean Esses parâmetros sendo comuns para todos os métodosget' são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando todos os SLAs

Recupere todos os dados sobre todos os SLAs e suas propriedades.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "sla.get",
  "params": {
    "output": "extend",
    "selectSchedule": ["period_from", "period_to"],
    "selectExcludedDowntimes": ["name", "period_from", "period_to"],
    "selectServiceTags": ["tag", "operator", "value"],
    "preservekeys": true
  },
  "auth": "85dd04b94cbfad794616eb923be13c71",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "1": {
      "slaid": "1",
      "name": "Database Uptime",
      "period": "1",

```



```

        "slo": "99.9995",
        "effective_date": "1672444800",
        "timezone": "America/Toronto",
        "status": "1",
        "description": "Provide excellent uptime for main SQL database engines.",
        "service_tags": [
            {
                "tag": "Database",
                "operator": "0",
                "value": "MySQL"
            },
            {
                "tag": "Database",
                "operator": "0",
                "value": "PostgreSQL"
            }
        ],
        "schedule": [
            {
                "period_from": "0",
                "period_to": "601200"
            }
        ],
        "excluded_downtimes": [
            {
                "name": "Software version upgrade rollout",
                "period_from": "1648760400",
                "period_to": "1648764900"
            }
        ]
    },
    "id": 1
}

```

Fonte

CSla:get() in ui/include/classes/api/services/CSla.php.

sla.getsli

Descrição

objeto sla.getsli(parâmetros do objeto)

Este método permite calcular os dados do Indicador de Nível de Serviço (SLI).

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros contendo o ID do SLA, períodos de relatório e, opcionalmente, os IDs dos serviços - para calcular o SLI.

Parâmetro	Tipo	Descrição
slaid (requerido)	string	IDs de serviços para os quais retornar informações de disponibilidade.
period_from	integer	Data de início (inclusive) para relatar o SLI.
Valores possíveis: timestamp.		

Parâmetro	Tipo	Descrição
period_to	integer	Data de término (exclusiva) para relatar o SLI.
periods	array	Valores possíveis: timestamp. Número preferencial de períodos a serem relatados.
serviceids	string/array	Valores possíveis: 1-100 IDs de serviços para os quais retornar o SLI.

Particionamento de períodos

O seguinte demonstra o arranjo de period slices retornados com base em combinações de parâmetros..

Parâmetros	Descrição
period_from period_to periods	
- - -	Os últimos 20 períodos (incluindo o atual), mas não após o primeiro período disponível com base na data de vigência do SLA.
- - specified	Os últimos períodos especificados pelo parâmetro periods .
- specified -	Os últimos 20 períodos antes da data especificada , mas não após o primeiro período disponível com base na data de vigência do SLA.
- specified specified	Os últimos períodos especificados pelo parâmetro periods antes da data especificada .
specified - -	Os primeiros 20 períodos (incluindo o atual), mas não após o atual.
specified - specified	Os primeiros períodos especificados pelo parâmetro periods começando com o specified date .
specified specified -	Períodos dentro do intervalo de datas especificado, mas não mais de 100 e não após o primeiro período disponível com base na data de vigência do SLA.
specified specified specified	Períodos dentro do intervalo de datas especificado, mas não mais do que o número especificado de períodos e não após o primeiro período disponível com base na data de vigência do SLA.

Valores de retorno

(object) Retorna os resultados do cálculo.

Propriedade	Tipo	Descrição
periods	array	Lista dos períodos relatados. Cada período relatado é representado como um objeto que consiste em: - period_from - Data de início do período relatado (timestamp). - period_to - Término data do período relatado (timestamp).
serviceids	array	Os períodos são classificados pelo campo period_from crescente. Lista de IDs de serviço nos períodos relatados.
sli	array	A ordem de classificação da lista não está definida. Mesmo que o parâmetro serviceids tenha sido passado para o método sla.getsli. Dados SLI (como uma array bidimensional) para cada período e serviço informado. O índice da propriedade periods é usado como a primeira dimensão do sli property. O índice da propriedade serviceids é usado como a segunda dimensão da propriedade sli.

Dados SLI

Os dados SLI retornados para cada período e serviço relatado consistem em:

Propriedade	Tipo	Descrição
uptime	integer	Quantidade de tempo que o serviço passou em um estado OK durante o tempo de atividade programado, menos os tempos de inatividade excluídos.

Propriedade	tipo	Descrição
downtime	integer	Quantidade de tempo que o serviço gastou em um estado não OK durante o tempo de atividade programado, menos os tempos de inatividade excluídos.
sli	float	SLI (por cento do tempo de atividade total), com base no tempo de atividade e tempo de inatividade.
error_budget	integer	Orçamento de erro (em segundos), com base no SLI e no SLO.
excluded_days	array of times	Array de tempos de inatividade excluídos neste período de relatório. Cada objeto conterá os seguintes parâmetros: - name - Nome do tempo de inatividade excluído. - period_from - Data e hora de início (inclusive) do tempo de inatividade excluído. - period_to - Data e hora de término (exclusiva) do tempo de inatividade excluído.
Os tempos de inatividade excluídos são classificados pelo campo period_from crescente.		

Exemplos

Calculando SLI

Recuperar SLI em serviços com IDs "50", "60" e "70" vinculados a um SLA com ID "5" por 3 períodos a partir de 01 de novembro de 2021.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.getsli",
  "params": {
    "slaid": "5",
    "serviceids": [
      50,
      60,
      70
    ],
    "periods": 3,
    "period_from": "1635724800"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "periods": [
      {
        "period_from": 1635724800,
        "period_to": 1638316800
      },
      {
        "period_from": 1638316800,
        "period_to": 1640995200
      },
      {
        "period_from": 1640995200,
        "period_to": 1643673600
      }
    ],
    "serviceids": [
      50,
      60,
      70
    ]
  }
}
```

```

],
"sli": [
  [
    {
      "uptime": 1186212,
      "downtime": 0,
      "sli": 100,
      "error_budget": 0,
      "excluded_downtimes": [
        {
          "name": "Excluded Downtime - 1",
          "period_from": 1637836212,
          "period_to": 1638316800
        }
      ]
    },
    {
      "uptime": 1186212,
      "downtime": 0,
      "sli": 100,
      "error_budget": 0,
      "excluded_downtimes": [
        {
          "name": "Excluded Downtime - 1",
          "period_from": 1637836212,
          "period_to": 1638316800
        }
      ]
    },
    {
      "uptime": 1186212,
      "downtime": 0,
      "sli": 100,
      "error_budget": 0,
      "excluded_downtimes": [
        {
          "name": "Excluded Downtime - 1",
          "period_from": 1637836212,
          "period_to": 1638316800
        }
      ]
    }
  ],
  [
    {
      "uptime": 1147548,
      "downtime": 0,
      "sli": 100,
      "error_budget": 0,
      "excluded_downtimes": [
        {
          "name": "Excluded Downtime - 1",
          "period_from": 1638439200,
          "period_to": 1639109652
        }
      ]
    },
    {
      "uptime": 1147548,
      "downtime": 0,
      "sli": 100,
      "error_budget": 0,
    }
  ]
]

```

```

        "excluded_downtimes": [
            {
                "name": "Excluded Downtime - 1",
                "period_from": 1638439200,
                "period_to": 1639109652
            }
        ]
    },
    {
        "uptime": 1147548,
        "downtime": 0,
        "sli": 100,
        "error_budget": 0,
        "excluded_downtimes": [
            {
                "name": "Excluded Downtime - 1",
                "period_from": 1638439200,
                "period_to": 1639109652
            }
        ]
    }
],
[
    {
        "uptime": 1674000,
        "downtime": 0,
        "sli": 100,
        "error_budget": 0,
        "excluded_downtimes": []
    },
    {
        "uptime": 1674000,
        "downtime": 0,
        "sli": 100,
        "error_budget": 0,
        "excluded_downtimes": []
    },
    {
        "uptime": 1674000,
        "downtime": 0,
        "sli": 100,
        "error_budget": 0,
        "excluded_downtimes": []
    }
]
],
{
    "id": 1
}

```

Fonte

CService::getSli() in ui/include/classes/api/services/CService.php.

sla.update

Descrição

object sla.update(object/array slaids)

Este método permite atualizar as entradas de SLA existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades de SLA a serem atualizadas.

A propriedade `slaid` deve ser definida para cada SLA, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das **propriedades de SLA padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
service_tags	array	SLA service_tags para substituir as atuais etiquetas de serviço SLA. Pelo menos uma etiqueta de serviço deve ser especificada. SLA schedule para substituir o atual.
schedule	array	Especificar o parâmetro como vazio será interpretado como uma programação 24x7.
excluded_downtimes	array	SLA excluded_downtimes para substituir os atuais.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos SLAs atualizados na propriedade `slaid`s.

Exemplos**Atualizando etiquetas de serviço**

Fazer SLA com ID "5" a ser calculado em intervalos mensais para serviços relacionados ao NoSQL, sem alterar sua programação ou excluir downtimes; defina SLO para 95%.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "sla.update",
  "params": [
    {
      "slaid": "5",
      "name": "NoSQL Database engines",
      "slo": "95",
      "period": 2,
      "service_tags": [
        {
          "tag": "Database",
          "operator": "0",
          "value": "Redis"
        },
        {
          "tag": "Database",
          "operator": "0",
          "value": "MongoDB"
        }
      ]
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "slais": [
      "5"
    ]
  },
  "id": 1
}
```

Alterando a programação de um SLA

Altere o SLA com ID "5" para uma programação 24 horas por dia, 7 dias por semana.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.update",
  "params": {
    "slaid": "5",
    "schedule": []
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "slais": [
      "5"
    ]
  },
  "id": 1
}
```

Alterando os tempos de inatividade excluídos para um SLA

Adicione um tempo de inatividade planejado de atualização de RAM de 4 horas em 6 de abril de 2022, mantendo (precisa ser definido novamente) uma atualização de software existente planejada em 4 de julho para o SLA com ID "5".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "service.update",
  "params": {
    "slaid": "5",
    "excluded_downtimes": [
      {
        "name": "Software version upgrade rollout",
        "period_from": "1648760400",
        "period_to": "1648764900"
      },
      {
        "name": "RAM upgrade",
        "period_from": "1649192400",
        "period_to": "1649206800"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "slais": [
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CSla::update() in ui/include/classes/api/services/CSla.php.

Tarefa

Essa classe foi projetada para trabalhar com tarefas (como verificar itens ou regras de descoberta de baixo nível sem recarregar a configuração).

Referências de objetos:

- [Task](#)
- ['Check now'](#)
- ['Diagnostic information'](#)
- [Statistic request object](#)
- [Statistic result object](#)

Métodos disponíveis:

- [task.create](#) - criando novas tarefas
- [task.get](#) - recuperando tarefas

> Objeto de tarefa

Os objetos a seguir estão diretamente relacionados à API task.

O objeto de tarefa tem as seguintes propriedades:

Propriedade	Type	Descrição
taskid	string	(somente leitura) ID da tarefa.
type (requerido)	integer	Tipo de tarefa. Valores possíveis: 1 - Informações de diagnóstico; 6 - Verifique agora.
status	integer	(somente leitura) Status da tarefa. Valores possíveis: 1 - nova tarefa; 2 - tarefa em andamento; 3 - tarefa concluída; 4 - tarefa expirou.
clock	timestamp	(somente leitura) Hora em que a tarefa foi criada.
ttl	integer	(somente leitura) O tempo em segundos após o qual a tarefa expira.
proxy_hostid	string	ID do proxy sobre o qual a estatística de informações de diagnóstico é coletada.
request (requerido)	object	Ignorado para tarefas 'Verificar agora'. Objeto de solicitação de tarefa de acordo com o tipo de tarefa: O objeto da tarefa 'Verificar agora' é descrito em detalhes abaixo ; Objeto de 'Informações de diagnóstico' tarefa é descrita em detalhes abaixo .

Propriedade	Type	Descrição
result	object	(somente leitura) Objeto de resultado da tarefa de informações de diagnóstico. Pode conter NULL se o resultado ainda não estiver pronto. O objeto de resultado é descrito em detalhes abaixo .

Objeto de solicitação 'Check now'

O objeto de solicitação de tarefa 'Check now' tem as seguintes propriedades.

Propriedade	Type	Descrição
itemid	string	ID do item e regras de descoberta de baixo nível.

Objeto de solicitação 'Diagnostic information'

O objeto de solicitação de tarefa de informações de diagnóstico tem as seguintes propriedades. O objeto de solicitação de estatística para todos os tipos de propriedades é [descrito em detalhes abaixo](#).

Propriedade	Type	Descrição
historycache	object	Solicitação de estatística de cache de histórico. Disponível no servidor e proxy.
valuecache	object	Solicitação de estatísticas de cache de itens. Disponível no servidor.
preprocessing	object	Solicitação de estatística do gerenciador de pré-processamento. Disponível no server e proxy.
alerting	object	Solicitação de estatística do gerenciador de alertas. Disponível no servidor.
lld	objeto	solicitação de estatística do gerenciador de LLC. Disponível no servidor.

Objeto de solicitação de estatística

O objeto de solicitação de estatística é usado para definir que tipo de informação deve ser coletada sobre os processos internos do servidor/proxy. Tem as seguintes propriedades.

Propriedade	Type	Descrição
stats	query	Propriedades do objeto estatístico a serem retornadas. A lista de campos disponíveis para cada tipo de estatística de informação de diagnóstico é descrita em detalhes abaixo .
top	object	<p>Padrão: extend retornará todos os campos de estatística disponíveis. Objeto para classificar e limitar os valores estatísticos retornados. A lista de campos disponíveis para cada tipo de estatística de informações de diagnóstico é descrita em detalhes abaixo.</p> <p>Exemplo: { "source.alerts": 10 }</p>

Lista de campos estatísticos disponíveis para cada tipo de solicitação de informações de diagnóstico

Os campos de estatística a seguir podem ser solicitados para cada tipo de propriedade de solicitação de informações de diagnóstico.

Tipo de diagnóstico	Campos disponíveis	Descrição
historycache	items values memory	Número de itens em cache. Número de valores em cache. Estatísticas de memória compartilhada (free space, number of used chunks, number of free chunks, max size of free chunk).

Tipo de diagnóstico	Campos disponíveis	Descrição
valuecache	memory.data	Estatísticas de memória compartilhada do cache de dados do histórico.
	memory.index	Estatísticas de memória compartilhada do cache de índice de histórico.
	items	Número de itens em cache.
	values	Número de valores em cache.
	memory	Estatísticas de memória compartilhada (free space, number of used chunks, number of free chunks, max size of free chunk).
preprocessing	mode	Modo de cache de valor.
	values	Número de valores enfileirados.
	preproc.values	Número de valores enfileirados com etapas de pré-processamento.
alerting	alerts	Número de alertas na fila.
lld	rules	Número de regras enfileiradas.
	values	Número de valores enfileirados.

Lista de campos de classificação disponíveis para cada tipo de solicitação de informações de diagnóstico

Os campos estatísticos a seguir podem ser usados para classificar e limitar as informações solicitadas.

Tipo de diagnóstico	Campos disponíveis	Tipo
historycache	values	integer
valuecache	values	integer
	request.values	integer
preprocessing	values	integer
alerting	media.alerts	integer
	source.alerts	integer
lld	values	integer

Objeto de resultado estatístico

O objeto de resultado da estatística é recuperado no campo `result` do objeto de tarefa.

Propriedade	Type	Descrição
status	integer	(somente leitura) Status do resultado da tarefa.
data	string/object	Valores possíveis: -1 - ocorreu um erro durante a execução da tarefa; 0 - o resultado da tarefa é criado.
		Resultados de acordo com o objeto de solicitação de estatística de determinada tarefa de informações de diagnóstico. Contém uma string de mensagem de erro se ocorreu um erro durante a execução da tarefa.

task.create

Descrição

`object task.create(object/array tasks)`

Este método permite criar uma nova tarefa (como coletar dados de diagnóstico ou verificar itens ou regras de descoberta de baixo nível sem recarregar a configuração).

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Veja [Usuário funções](#) para mais informações.

Parâmetros

(object/array) Uma tarefa para criar.

O método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
type (requerido)	integer	Tipo de tarefa. Valores possíveis: 1 - Informações de diagnóstico; 6 - Verifique agora.
request (requerido)	object	Objeto de solicitação de tarefa de acordo com o tipo de tarefa. O formato correto do objeto de solicitação é descrito na seção Task object .
proxy_hostid	integer	Proxy sobre qual tarefa de informações de diagnóstico coletará dados. Ignorado para tarefas "Verificar agora".

Observe que as tarefas 'Verificar agora' podem ser criadas apenas para os seguintes tipos de itens/regras de descoberta:

- Zabbix agent
- SNMPv1/v2/v3 agent
- Simple check
- Internal check
- External check
- Database monitor
- HTTP agent
- IPMI agent
- SSH agent
- TELNET agent
- Calculated check
- JMX agent

Valores de retorno

(object) Retorna um objeto contendo os IDs das tarefas criadas na propriedade taskids. Uma tarefa é criada para cada item e regra de descoberta de baixo nível. A ordem dos IDs retornados corresponde à ordem dos itemids passados.

Exemplos**Criando uma tarefa**

Crie uma tarefa `check now` para dois itens. Um é um item, o outro é uma regra de descoberta de baixo nível.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "task.create",
  "params": [
    {
      "type": 6,
      "request": {
        "itemid": "10092"
      }
    },
    {
      "type": "6",
      "request": {
        "itemid": "10093"
      }
    }
  ]
}
```

```

    }
  },
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "taskids": [
      "1",
      "2"
    ]
  },
  "id": 1
}

```

Create a task diagnostic information task.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "task.create",
  "params": [
    {
      "type": 1,
      "request": {
        "alerting": {
          "stats": [
            "alerts"
          ],
          "top": {
            "media.alerts": 10
          }
        },
        "l1d": {
          "stats": "extend",
          "top": {
            "values": 5
          }
        }
      },
      "proxy_hostid": 0
    }
  ],
  "auth": "700ca65537074ec963db7efabda78259",
  "id": 2
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": {
    "taskids": [
      "3"
    ]
  },
  "id": 2
}

```

Veja também

- [Task](#)
- ['Check now'](#)
- ['Diagnostic information'](#)
- [Statistic request object](#)

Fonte

CTask::create() in ui/include/classes/api/services/CTask.php.

task.get

Descrição

integer/array task.get(object parameters)

O método permite recuperar tarefas de acordo com os parâmetros fornecidos.

O método retorna detalhes apenas sobre tarefas de 'informações de diagnóstico'.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Type	Descrição
taskids	string/array	Retornar apenas tarefas com os IDs fornecidos.
output	query	Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
preservekeys	boolean	

Valores de retorno

(integer/array) Retorna um array de objetos.

Exemplos

Recuperar tarefa por ID

Recupere todos os dados sobre a tarefa com o ID "1". Request:

```
{
  "jsonrpc": "2.0",
  "method": "task.get",
  "params": {
    "output": "extend",
    "taskids": "1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "taskid": "1",
      "type": "7",
      "status": "3",
      "clock": "1601039076",
      "ttl": "3600",

```

```

        "proxy_hostid": null,
        "request": {
            "alerting": {
                "stats": [
                    "alerts"
                ],
                "top": {
                    "media.alerts": 10
                }
            },
            "lld": {
                "stats": "extend",
                "top": {
                    "values": 5
                }
            }
        },
        "result": {
            "data": {
                "alerting": {
                    "alerts": 0,
                    "top": {
                        "media.alerts": []
                    }
                },
                "time": 0.000663
            },
            "lld": {
                "rules": 0,
                "values": 0,
                "top": {
                    "values": []
                }
            },
            "time": 0.000442
        },
        "status": "0"
    }
},
    ],
    "id": 1
}

```

Veja também

- [Task](#)
- [Statistic result object](#)

Fonte

CTask::get() in ui/include/classes/api/services/CTask.php.

Template

Essa classe é desenhada para trabalhar com templates.

Referências de objeto:

- [Template](#)

Métodos disponíveis:

- [template.create](#) - criar novos templates
- [template.delete](#) - deletar templates
- [template.get](#) - recuperar templates
- [template.massadd](#) - adicionar objetos relacionados aos templates
- [template.massremove](#) - remover objetos relacionados aos templates

- **template.massupdate** - substituir ou remover objetivos relacionados dos templates
- **template.update** - atualizar templates

> Objeto Template

Os objetos a seguir estão diretamente relacionados à API `template`.

Template

O objeto `template` tem as seguintes propriedades.

Propriedade	Tipo	Descrição
<code>templateid</code>	string	(somente leitura) ID do <code>template</code> .
host (obrigatório)	string	Nome técnico do <code>template</code> .
<code>descricao</code>	text	Descrição do <code>template</code> .
<code>nome</code>	string	Nome visível do <code>template</code> .
<code>uuid</code>	string	Padrão: valor da propriedade <code>dohost</code> . Identificador único universal, usado para vincular templates importados a templates já existentes. Gerado automaticamente, se não fornecido.
Para operações de atualização, esse campo é somente leitura.		

Observe que, para alguns métodos (atualizar, excluir), a combinação de parâmetros obrigatórios/opcionais é diferente.

Etiqueta de `template`

O objeto de `tag` de modelo possui as seguintes propriedades.

Propriedade	Tipo	Descrição
etiqueta (obrigatório)	string	Nome da etiqueta do <code>template</code> .
<code>valor</code>	string	Valor da etiqueta do <code>template</code> .

template.create

Descrição

`object template.create(object/array templates)`

Este método permite criar novos modelos.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(`object/array`) Modelos para criar.

Além das **propriedades de modelo padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Type	Descrição
groups (requerido)	object/array	Host grupos para adicionar o modelo. Os grupos de host devem ter a propriedade <code>groupid</code> definida.
<code>tags</code>	object/array	Modelo etiquetas .
<code>templates</code>	object/array	Modelos a serem vinculados ao modelo. Os modelos devem ter a propriedade <code>templateid</code> definida.

Parâmetro	Type	Descrição
macros	object/array	Macros do usuário a serem criadas para o modelo.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos templates criados na propriedade `templateids`. A ordem dos IDs retornados corresponde à ordem dos modelos passados.

Exemplos

Criando um modelo

Crie um modelo com etiquetas e vincule dois modelos a este modelo.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "template.create",
  "params": {
    "host": "Linux template",
    "groups": {
      "groupid": 1
    },
    "templates": [
      {
        "templateid": "11115"
      },
      {
        "templateid": "11116"
      }
    ],
    "tags": [
      {
        "tag": "Host name",
        "value": "{HOST.NAME}"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "11117"
    ]
  },
  "id": 1
}
```

Fonte

CTemplate::create() in ui/include/classes/api/services/CTemplate.php.

template.delete

Descrição

object `template.delete(array templateIds)`

Este método permite excluir templates.

A exclusão de um template causará a exclusão de todas as entidades do modelo (itens, triggers, gráficos, etc.). Para deixar entidades de modelo com os hosts, mas excluir o próprio modelo, primeiro desvincule o modelo dos hosts necessários usando um destes métodos: `template.update`, `template.massupdate`, `host.update`, `host.massupdate`.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos templates a serem deletados.

Retorno de valores

(object) Retorna um objeto contendo os IDs dos templates deletados na propriedade `templateids`.

Exemplos

Deletando múltiplos templates

Deleta dois templates..

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.delete",
  "params": [
    "13",
    "32"
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "13",
      "32"
    ]
  },
  "id": 1
}
```

Fonte

CTemplate::delete() in ui/include/classes/api/services/CTemplate.php.

template.get

Descrição

integer/array `template.get(object parameters)`

O método permite recuperar modelos de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
templateids	string/array	Retorna apenas modelos com os IDs de modelo fornecidos.
groupids	string/array	Retorna apenas modelos que pertencem aos grupos de hosts fornecidos.
parentTemplateids	string/array	Retorne apenas modelos que são pais dos modelos fornecidos.
hostids	string/array	Retorne apenas modelos vinculados aos hosts/modelos fornecidos.
graphids	string/array	Retorne apenas modelos que contenham os gráficos fornecidos.
itemids	string/array	Retorne apenas modelos que contenham os itens fornecidos.
triggerids	string/array	Retorne apenas modelos que contenham os triggers fornecidas.
with_items	flag	Retorne apenas templates que tenham itens.
with_triggers	flag	Retorne apenas templates que tenham triggers.
with_graphs	flag	Retorne apenas templates que tenham gráficos.
with_httptests	flag	Retorne apenas templates que tenham cenários da web.
evaltype	integer	Regras para pesquisa de tags.
		Valores possíveis: 0 - (padrão) e/ou; 2 - ou.
tags	array/object	Retorna apenas modelos com tags fornecidas. Correspondência exata por tag e pesquisa com ou sem distinção entre maiúsculas e minúsculas por valor de tag, dependendo do valor do operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Uma matriz vazia retorna todos os templates.
		Possíveis valores do operador: 0 - (padrão) Contém; 1 - Igual; 2 - Não contém; 3 - Diferente; 4 - Existe; 5 - Não existe.
selectGroups	query	Retorne os grupos de hosts aos quais o modelo pertence na propriedade grupos property.
selectTags	query	Retorne tags de modelo na propriedade tags property.
selectHosts	query	Retorne os hosts que estão vinculados ao modelo na propriedade hosts property.
selectTemplates	query	Suporta count. Retorne templates para os quais o modelo é filho, na propriedade templates .
selectParentTemplates	query	Suporta count. Retorne templates para os quais o modelo é pai, na propriedade parentTemplates .
selectHttpTests	query	Suporta count. Retorne os cenários da web do modelo na propriedade httpTests .
selectItems	query	Suporta count. Retorne itens do modelo na propriedade items .
selectDiscoveries	query	Suporta count. Retorne descobertas de baixo nível do modelo na propriedade discoveries .
selectTriggers	query	Suporta count. Retorne os triggers de templates na propriedade de triggers .
selectGraphs	query	Suporta count. Retorne as triggers de templates na propriedade graphs .
		Suporta count.

Parâmetro	Tipo	Descrição
selectMacros	query	Retorne as macros do modelo na propriedade macros.
selectDashboards	query	Retorne os painéis do modelo na propriedade de dashboards .
selectValueMaps	query	Suporta count. Retorne uma propriedade valuemaps com mapas de valores de modelo.
limitSelects	integer	Limita o número de registros retornados por subseleções.
sortfield	string/array	Aplica-se às seguintes subseleções: selectTemplates - os resultados serão classificados por name; selectHosts - classificado por host; selectParentTemplates - classificado por host; selectItems - classificado por name; selectDiscoveries - classificado por name; selectTriggers - classificado por description; selectGraphs - classificado por name; selectDashboards - classificado por name. Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: hostid, host, name, status. Esses parâmetros sendo comuns para todos os métodos get que são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna ou:

- uma matriz de objetos;
- contagem de objetos recuperados, se o countOutput parâmetro tiver sido usado.

Exemplos

Recuperando templates por nome

Recupere todos os dados sobre dois templates chamados "Linux" e "Windows".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.get",
  "params": {
    "output": "extend",
    "filter": {
      "host": [
        "Linux",
        "Windows"
      ]
    }
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "proxy_hostid": "0",
      "host": "Linux",
      "status": "3",
      "disable_until": "0",
      "error": "",
      "available": "0",
      "errors_from": "0",
      "lastaccess": "0",
      "ipmi_authtype": "0",
      "ipmi_privilege": "2",
      "ipmi_username": "",
      "ipmi_password": "",
      "ipmi_disable_until": "0",
      "ipmi_available": "0",
      "snmp_disable_until": "0",
      "snmp_available": "0",
      "maintenanceid": "0",
      "maintenance_status": "0",
      "maintenance_type": "0",
      "maintenance_from": "0",
      "ipmi_errors_from": "0",
      "snmp_errors_from": "0",
      "ipmi_error": "",
      "snmp_error": "",
      "jmx_disable_until": "0",
      "jmx_available": "0",
      "jmx_errors_from": "0",
      "jmx_error": "",
      "name": "Linux",
      "flags": "0",
      "templateid": "10001",
      "description": "",
      "tls_connect": "1",
      "tls_accept": "1",
      "tls_issuer": "",
      "tls_subject": "",
      "tls_psk_identity": "",
      "tls_psk": "",
      "uuid": "282ffe33afc74cccaf1524d9aa9dc502"
    },
    {
      "proxy_hostid": "0",
      "host": "Windows",
      "status": "3",
      "disable_until": "0",
      "error": "",
      "available": "0",
      "errors_from": "0",
      "lastaccess": "0",
      "ipmi_authtype": "0",
      "ipmi_privilege": "2",
      "ipmi_username": "",
      "ipmi_password": "",
      "ipmi_disable_until": "0",
      "ipmi_available": "0",
      "snmp_disable_until": "0",
      "snmp_available": "0",

```

```

        "maintenanceid": "0",
        "maintenance_status": "0",
        "maintenance_type": "0",
        "maintenance_from": "0",
        "ipmi_errors_from": "0",
        "snmp_errors_from": "0",
        "ipmi_error": "",
        "snmp_error": "",
        "jmx_disable_until": "0",
        "jmx_available": "0",
        "jmx_errors_from": "0",
        "jmx_error": "",
        "name": "Windows",
        "flags": "0",
        "templateid": "10081",
        "description": "",
        "tls_connect": "1",
        "tls_accept": "1",
        "tls_issuer": "",
        "tls_subject": "",
        "tls_psk_identity": "",
        "tls_psk": "",
        "uuid": "522d17e1834049be879287b7c0518e5d"
    }
],
    "id": 1
}

```

Retrieving hosts by template

Retrieve hosts that have the "10001" (Linux by Zabbix agent) template linked to them.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "template.get",
    "params": {
        "output": "templateid",
        "templateids": "10001",
        "selectHosts": ["hostid", "name"]
    },
    "auth": "038e1d7b1735c6a5436ee9eae095879e",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": [
        {
            "templateid": "10001",
            "hosts": [
                {
                    "hostid": "10084",
                    "name": "Zabbix server"
                },
                {
                    "hostid": "10603",
                    "name": "Host 1"
                },
                {
                    "hostid": "10604",
                    "name": "Host 2"
                }
            ]
        }
    ]
}

```

```

    }
  ]
},
"id": 1
}

```

Pesquisando por tags de templates

Recupere modelos que tenham a tag "Host name" igual a "{HOST.NAME}".

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "template.get",
  "params": {
    "output": ["hostid"],
    "selectTags": "extend",
    "evaltype": 0,
    "tags": [
      {
        "tag": "Host name",
        "value": "{HOST.NAME}",
        "operator": 1
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostid": "10402",
      "tags": [
        {
          "tag": "Host name",
          "value": "{HOST.NAME}"
        }
      ]
    }
  ],
  "id": 1
}

```

Veja também

- [Grupo de hosts](#)
- [Template](#)
- [Macro de usuário](#)
- [interface do host](#)

Fonte

CTemplate::get() in ui/include/classes/api/services/CTemplate.php.

template.massadd

Descrição

object template.massadd(object parameters)

Este método permite adicionar simultaneamente vários objetos relacionados aos modelos fornecidos.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros contendo os IDs dos modelos a serem atualizados e os objetos a serem adicionados aos modelos.

O método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
templates (obrigatório)	object/array	Templates a serem atualizados.
grupos	object/array	Os templates devem ter a propriedade <code>templateid</code> definida. Grupos de hosts aos quais adicionar os modelos fornecidos.
macros	object/array	Os grupos de hosts devem ter a propriedade <code>groupid</code> definida. Macros de usuário a serem criadas para os templates fornecidos.
templates_link	object/array	Templates para vincular aos templates fornecidos.
		Os templates devem ter a <code>templateid</code> propriedade definida.

Retorno de valores

(object) Retorna um objeto que contenha os IDs dos templates atualizados na propriedade `templateids`.

Exemplos

Vincular um grupo a modelos

Adicione o grupo de hosts "2" a dois templates.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.massadd",
  "params": {
    "templates": [
      {
        "templateid": "10085"
      },
      {
        "templateid": "10086"
      }
    ],
    "groups": [
      {
        "groupid": "2"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10085",
      "10086"
    ]
  }
}
```

```

    ],
    "id": 1
}

```

Vincular dois templates a um template

Vincule os templates "10106" e "10104" ao modelo.

Requisição:

```

{
  "jsonrpc": "2.0",
  "method": "template.massadd",
  "params": {
    "templates": [
      {
        "templateid": "10073"
      }
    ],
    "templates_link": [
      {
        "templateid": "10106"
      },
      {
        "templateid": "10104"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}

```

Resposta:

```

{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10073"
    ]
  },
  "id": 1
}

```

Veja também

- [template.update](#)
- [Host](#)
- [Grupo de Host](#)
- [Macro de usuário](#)

Fonte

CTemplate::massAdd() in ui/include/classes/api/services/CTemplate.php.

template.massremove

Descrição

`object template.massremove(object parameters)`

Este método permite remover objetos relacionados de vários templates.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [função do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros contendo os IDs dos templates a serem atualizados e os objetos que devem ser removidos.

Parâmetro	Tipo	Descrição
templateids (required)	string/array	IDs dos templates a serem atualizados.
groupids	string/array	Grupos de hosts dos quais remover os templates fornecidos.
macros	string/array	Macros de usuário para excluir dos templates fornecidos.
templateids_clear	string/array	Templates para desvincular e limpar dos templates fornecidos (upstream).
templateids_link	string/array	Templates para desvincular dos templates fornecidos (upstream).

Valores de retorno

(object) Retorna um objeto contendo os IDs dos modelos atualizados na propriedade `templateids`.

Exemplos

Removendo templates de um grupo

Remover 2 templates do grupo "2".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.massremove",
  "params": {
    "templateids": [
      "10085",
      "10086"
    ],
    "groupids": "2"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10085",
      "10086"
    ]
  },
  "id": 1
}
```

Desvinculando templates de um host

Desvincule os templates "10106", "10104" do template "10085".

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.massremove",
  "params": {
    "templateids": "10085",
    "templateids_link": [
      "10106",
      "10104"
    ]
  },
  "id": 1
}
```

```
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10085"
    ]
  },
  "id": 1
}
```

Veja também

- [template.update](#)
- [macro de usuário](#)

Fonte

CTemplate::massRemove() in ui/include/classes/api/services/CTemplate.php.

template.massupdate

Descrição

`object template.massupdate(object parameters)`

Este método permite substituir ou remover simultaneamente objetos relacionados e atualizar propriedades em vários modelos.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros contendo os IDs dos templates a serem atualizados e os objetos a serem substituídos pelos templates.

O método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
templates (obrigatório)	object/array	Modelos a serem atualizados.
grupos	object/array	Os modelos devem ter a propriedade <code>templateid</code> definida. Grupos de hosts para substituir os grupos de hosts atuais aos quais os templates pertencem.
macros	object/array	Os grupos de hosts devem ter a propriedade <code>groupid</code> definida. Macros de usuário para substituir as macros de usuário atuais nos templatesa fornecidos.
templates_clear	object/array	Templates para desvincular e limpar dos templates fornecidos.
templates_link	object/array	Os templates devem ter a propriedade <code>templateid</code> definida. Templates para substituir os templates atualmente vinculados.
		Os templates devem ter a propriedade <code>templateid</code> definida.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos modelos atualizados na propriedade `templateids`.

Exemplos

Substituindo grupos de hosts

Desvincule e limpe o modelo "10091" dos templates fornecidos.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "template.massupdate",
  "params": {
    "templates": [
      {
        "templateid": "10085"
      },
      {
        "templateid": "10086"
      }
    ],
    "templates_clear": [
      {
        "templateid": "10091"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10085",
      "10086"
    ]
  },
  "id": 1
}
```

Veja também

- [template.update](#)
- [template.massadd](#)
- [Grupo de hosts](#)
- [Macro de usuário](#)

Fonte

CTemplate::massUpdate() in ui/include/classes/api/services/CTemplate.php.

template.update

Description

object template.update(object/array templates)

This method allows to update existing templates.

Note:

This method is only available to Admin and Super admin user types. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parameters

(object/array) Template properties to be updated.

The `templateid` property must be defined for each template, all other properties are optional. Only the given properties will be updated, all others will remain unchanged.

Additionally to the **standard template properties**, the method accepts the following parameters.

Parameter	Type	Description
groups	object/array	Host groups to replace the current host groups the templates belong to. The host groups must have the <code>groupid</code> property defined.
tags	object/array	Template tags to replace the current template tags.
hosts	object/array	Hosts and templates to replace the ones the templates are currently linked to. Both hosts and templates must use the <code>hostid</code> property to pass an ID.
macros	object/array	User macros to replace the current user macros on the given templates.
templates	object/array	Templates to replace the currently linked templates. Templates that are not passed are only unlinked.
templates_clear	object/array	The templates must have the <code>templateid</code> property defined. Templates to unlink and clear from the given templates. The templates must have the <code>templateid</code> property defined.

Return values

(object) Returns an object containing the IDs of the updated templates under the `templateids` property.

Exemplos

Renaming a template

Rename the template to "Template OS Linux".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "template.update",
  "params": {
    "templateid": "10086",
    "name": "Template OS Linux"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "templateids": [
      "10086"
    ]
  },
  "id": 1
}
```

Updating template tags

Replace all template tags with a new one.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "template.update",
  "params": {
    "templateid": "10086",
    "tags": [
      {
        "tag": "Host name",
        "value": "{HOST.NAME}"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "hostids": [
      "10086"
    ]
  },
  "id": 1
}
```

Fonte

CTemplate::update() in ui/include/classes/api/services/CTemplate.php.

Tendência

Esta classe foi projetada para trabalhar com dados de tendência.

Referências de objetos:

- [Trend](#)

Métodos disponíveis:

- [trend.get](#) - recuperando tendências

> Objeto de tendência

Os objetos a seguir estão diretamente relacionados à API `trend`.

Note:

Os objetos de tendência diferem dependendo do tipo de informação do item. Eles são criados pelo servidor Zabbix e não podem ser modificados via API.

Tendência flutuante

O objeto de tendência flutuante tem as seguintes propriedades.

Propriedade	Type	Descrição
clock	timestamp	Timestamp de uma hora para a qual o valor foi calculado. Por exemplo, timestamp de 04:00:00 significa valores calculados para o período 04:00:00-04:59:59.
itemid	integer	ID do item relacionado.
num	integer	Número de valores que estavam disponíveis para a hora.
value_min	float	Valor mínimo por hora.
value_avg	float	Valor médio por hora.
value_max	float	Valor máximo por hora.

Tendência de números inteiros

O objeto de tendência inteiro tem as seguintes propriedades.

Propriedade	Type	Descrição
clock	timestamp	Timestamp de uma hora para a qual o valor foi calculado. Por exemplo, timestamp de 04:00:00 significa valores calculados para o período 04:00:00-04:59:59.
itemid	integer	ID do item relacionado.
num	integer	Número de valores que estavam disponíveis para a hora.
value_min	integer	Valor mínimo por hora.
value_avg	integer	Valor médio por hora.
value_max	integer	Valor máximo por hora.

trend.get

Descrição

`integer/array trend.get(object parameters)`

O método permite recuperar dados de tendência de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Type	Descrição
itemids	string/array	Retornar apenas tendências com os IDs de item fornecidos.
time_from	timestamp	Retorna apenas valores que foram coletados após ou no momento determinado.
time_till	timestamp	Retorna apenas valores que foram coletados antes ou no momento determinado.
countOutput	boolean	Conte o número de objetos recuperados.
limit	integer	Limite a quantidade de objetos recuperados.
output	query	Definir campos para saída.

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando dados de tendência do item

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trend.get",
  "params": {
    "output": [
      "itemid",
      "clock",
      "num",
      "value_min",
      "value_avg",
      "value_max",
    ],
    "itemids": [
      "23715"
    ],
    "limit": "1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "23715",
      "clock": "1446199200",
      "num": "60",
      "value_min": "0.165",
      "value_avg": "0.2168",
      "value_max": "0.35",
    }
  ],
  "id": 1
}
```

Fonte

CTrend::get() in ui/include/classes/api/services/CTrend.php.

Tipo de mídia

Esta classe foi projetada para trabalhar com tipos de mídia.

Referências de objetos:

- [Media type](#)

Métodos disponíveis:

- [mediatype.create](#) - criando novos tipos de mídia
- [mediatype.delete](#) - excluindo tipos de mídia
- [mediatype.get](#) - recuperando tipos de mídia
- [mediatype.update](#) - atualizando tipos de mídia

> Objeto de tipo de mídia

Os objetos a seguir estão diretamente relacionados à API `mediatype`.

Tipo de mídia

O objeto de tipo de mídia tem as seguintes propriedades.

Propriedade	Tipo	Descrição
<code>mediatypeid</code>	string	(somente leitura) ID do tipo de mídia.
<code>name</code> (requerido)	string	Nome do tipo de mídia.
<code>type</code> (requerido)	integer	Transporte usado pelo tipo de mídia. Valores possíveis: 0 - email; 1 - script; 2 - SMS; 4 - Webhook.
<code>exec_path</code>	string	Para tipos de mídia de script, <code>exec_path</code> contém o nome do script executado.
<code>gsm_modem</code>	string	Obrigatório para tipos de mídia de script. Nome do dispositivo serial do modem GSM.
<code>smtp_email</code>	string	Obrigatório para tipos de mídia SMS. Endereço de e-mail do qual as notificações serão enviadas.
<code>smtp_helo</code>	string	Obrigatório para tipos de mídia de e-mail. SMTP HELO.
<code>smtp_server</code>	string	Obrigatório para tipos de mídia de e-mail. SMTP server.
<code>smtp_port</code> <code>smtp_security</code>	integer integer	Obrigatório para tipos de mídia de e-mail. Porta do servidor SMTP para se conectar. Nível de segurança de conexão SMTP a ser usado. Valores possíveis: 0 - Nenhum; 1 - STARTTLS; 2 - SSL/TLS.

Propriedade	Tipo	Descrição
smtp_verify_host	integer	Host de verificação SSL para SMTP.
smtp_verify_peer	integer	Valores possíveis: 0 - Não; 1 - Sim. SSL verificar peer para SMTP.
smtp_authentication	integer	Valores possíveis: 0 - Não; 1 - Sim. Método de autenticação SMTP a ser usado.
status	integer	Valores possíveis: 0 - Nenhum; 1 - Senha normal. Se o tipo de mídia está ativado.
username	string	Valores possíveis: 0 - (padrão) ativado; 1 - desativado. User name.
exec_params	string	Usado para tipos de mídia de e-mail. Parâmetros de script.
maxsessions	integer	Cada parâmetro termina com um novo feed de linha. O número máximo de alertas que podem ser processados em paralelo.
maxattempts	integer	Valores possíveis para SMS: 1 - (padrão) Valores possíveis para outros tipos de mídia: 0 -100 O número máximo de tentativas para enviar um alerta.
attempt_interval	string	Valores possíveis: 1-100 Valor padrão: 3 O intervalo entre as tentativas de repetição. Aceita segundos e unidade de tempo com sufixo.
content_type	integer	Valores possíveis: 0-1h Valor padrão: 10s Formato da mensagem.
script	string	Valores possíveis: 0 - texto simples; 1 - (padrão) html. Corpo de javascript do script de webhook do tipo de mídia.
timeout	string	Tempo limite do script de webhook do tipo de mídia. Aceita segundos e unidade de tempo com sufixo. Valores possíveis: 1-60s Valor padrão: 30s

Propriedade	Tipo	Descrição
process_tags	integer	Define se a resposta do script do webhook deve ser interpretada como tags e essas tags devem ser adicionadas ao evento associado.
show_event_menu	integer	Valores possíveis: 0 - (padrão) Ignorar resposta do script do webhook. 1 - Processar a resposta do script do webhook como tags. Mostrar entrada de tipo de mídia na propriedade <code>problem.get</code> e <code>event.get</code> urls.
event_menu_url	string	Valores possíveis: 0 - (padrão) Não adicionar entrada urls. >1 - Adicionar tipo de mídia à propriedade urls. Defina a propriedade url da entrada do tipo de mídia na propriedade urls de <code>problem.get</code> e <code>event.get</code> .
event_menu_name	string	Defina a propriedade name da entrada do tipo de mídia na propriedade urls de <code>problem.get</code> e <code>event.get</code> .
parameters	array	Array de parâmetros de entrada do webhook .
description	string	Descrição do tipo de mídia.

Parâmetros do webhook

Os parâmetros passados para o script de webhook quando ele é chamado têm as seguintes propriedades.

Propriedade	Tipo	Descrição
name (requerido)	string	Nome do parâmetro.
value	string	Valor do parâmetro, macros de suporte. Macros compatíveis descritas na página .

Modelo de mensagem

O objeto de modelo de mensagem define um modelo que será usado como mensagem padrão para operações de ação para enviar uma notificação. Tem as seguintes propriedades.

Propriedade	Tipo	Descrição
eventsource (requerido)	integer	Origem do evento. Valores possíveis: 0 - acionadores; 1 - descoberta; 2 - registro automático; 3 - interno; 4 - serviços.
recovery (requerido)	integer	Modo de operação. Valores possíveis: 0 - operações; 1 - operações de recuperação; 2 - operações de atualização.
subject	string	Assunto da mensagem.
message	string	Mensagem de texto.

mediatype.create

Descrição

```
object mediatype.create(object/array mediaTypes)
```

Este método permite criar novos tipos de mídia.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver

[Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Tipos de mídia para criar.

Além das **propriedades de tipo de mídia padrão**, o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
parameters	array	Parâmetros do webhook a serem criados para o tipo de mídia.
message_templates	array	Modelos de mensagem a serem criados para o tipo de mídia.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tipos de mídia criados na propriedade `mediatypeids`. A ordem dos IDs retornados corresponde à ordem dos tipos de mídia passados

Exemplos

Criando um tipo de mídia de e-mail

Crie um novo tipo de mídia de e-mail com uma porta SMTP personalizada e modelos de mensagem.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "mediatype.create",
  "params": {
    "type": "0",
    "name": "E-mail",
    "smtp_server": "mail.example.com",
    "smtp_helo": "example.com",
    "smtp_email": "zabbix@example.com",
    "smtp_port": "587",
    "content_type": "1",
    "message_templates": [
      {
        "eventsourcing": "0",
        "recovery": "0",
        "subject": "Problem: {EVENT.NAME}",
        "message": "Problem \"{EVENT.NAME}\" on host \"{HOST.NAME}\" started at {EVENT.TIME}."
      },
      {
        "eventsourcing": "0",
        "recovery": "1",
        "subject": "Resolved in {EVENT.DURATION}: {EVENT.NAME}",
        "message": "Problem \"{EVENT.NAME}\" on host \"{HOST.NAME}\" has been resolved at {EVENT.TIME}."
      },
      {
        "eventsourcing": "0",
        "recovery": "2",
        "subject": "Updated problem in {EVENT.AGE}: {EVENT.NAME}",
        "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem \"{EVENT.NAME}\" on host \"{HOST.NAME}\""
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "mediatypeids": [
      "7"
    ]
  },
  "id": 1
}
```

Criando um tipo de mídia de script

Crie um novo tipo de mídia de script com um valor personalizado para o número de tentativas e o intervalo entre elas.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "mediatype.create",
  "params": {
    "type": "1",
    "name": "Push notifications",
    "exec_path": "push-notification.sh",
    "exec_params": "{ALERT.SENDTO}\n{ALERT.SUBJECT}\n{ALERT.MESSAGE}\n",
    "maxattempts": "5",
    "attempt_interval": "11s"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "mediatypeids": [
      "8"
    ]
  },
  "id": 1
}
```

Criando um tipo de mídia de webhook

Crie um novo tipo de mídia de webhook.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "mediatype.create",
  "params": {
    "type": "4",
    "name": "Webhook",
    "script": "var Webhook = {\r\n    token: null,\r\n    to: null,\r\n    subject: null,\r\n    messa",
    "parameters": [
      {
        "name": "Message",
        "value": "{ALERT.MESSAGE}"
      },
      {
        "name": "Subject",
        "value": "{ALERT.SUBJECT}"
      },
      {
        "name": "To",

```

```

        "value": "{ALERT.SENDTO}"
    },
    {
        "name": "Token",
        "value": "<Token>"
    }
]
},
"auth": "038e1d7b1735c6a5436ee9eae095879e",
"id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "mediatypeids": [
            "9"
        ]
    },
    "id": 1
}

```

Fonte

CMediaType::create() in ui/include/classes/api/services/CMediaType.php.

mediatype.delete

Descrição

object mediatype.delete(array mediaTypeIds)

Este método permite excluir tipos de mídia.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(array) IDs dos tipos de mídia a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tipos de mídia excluídos na propriedade mediatypeids.

Exemplos

Excluindo vários tipos de mídia

Exclua dois tipos de mídia.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "mediatype.delete",
    "params": [
        "3",
        "5"
    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "mediatypeids": [
      "3",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CMediaType::delete() in ui/include/classes/api/services/CMediaType.php.

mediatype.get

Descrição

integer/array mediatype.get(object parameters)

O método permite recuperar tipos de mídia de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método suporta os seguintes parâmetros.

Parâmetro	Tipo	Descrição
mediatypeids	string/array	Retorne apenas os tipos de mídia com os IDs fornecidos.
mediaids	string/array	Retorna apenas os tipos de mídia usados pela mídia fornecida.
userid	string/array	Retornar apenas os tipos de mídia usados por determinados usuários.
selectMessageTemplates	query	Retorna uma propriedade message_templates com uma matriz de mensagens de tipo de mídia.
selectUsers	query	Retornar uma propriedade users com os usuários que usam o tipo de mídia.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: mediatypeid. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes em comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;

- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando tipos de mídia

Recupere todos os tipos de mídia configurados

Request:

```
{
  "jsonrpc": "2.0",
  "method": "mediatype.get",
  "params": {
    "output": "extend",
    "selectMessageTemplates": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "mediatypeid": "1",
      "type": "0",
      "name": "Email",
      "smtp_server": "mail.example.com",
      "smtp_helo": "example.com",
      "smtp_email": "zabbix@example.com",
      "exec_path": "",
      "gsm_modem": "",
      "username": "",
      "passwd": "",
      "status": "0",
      "smtp_port": "25",
      "smtp_security": "0",
      "smtp_verify_peer": "0",
      "smtp_verify_host": "0",
      "smtp_authentication": "0",
      "exec_params": "",
      "maxsessions": "1",
      "maxattempts": "3",
      "attempt_interval": "10s",
      "content_type": "0",
      "script": "",
      "timeout": "30s",
      "process_tags": "0",
      "show_event_menu": "1",
      "event_menu_url": "",
      "event_menu_name": "",
      "description": "",
      "message_templates": [
        {
          "eventsourcing": "0",
          "recovery": "0",
          "subject": "Problem: {EVENT.NAME}",
          "message": "Problem started at {EVENT.TIME} on {EVENT.DATE}\r\nProblem name: {EVENT.NAME}"
        },
        {
          "eventsourcing": "0",
          "recovery": "1",
          "subject": "Resolved: {EVENT.NAME}",

```

```

        "message": "Problem has been resolved at {EVENT.RECOVERY.TIME} on {EVENT.RECOVERY.DATE}
    },
    {
        "eventsourc": "0",
        "recovery": "2",
        "subject": "Updated problem: {EVENT.NAME}",
        "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem at {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}",
    },
    {
        "eventsourc": "1",
        "recovery": "0",
        "subject": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}",
        "message": "Discovery rule: {DISCOVERY.RULE.NAME}\r\n\r\nDevice IP: {DISCOVERY.DEVICE.IPADDRESS}",
    },
    {
        "eventsourc": "2",
        "recovery": "0",
        "subject": "Autoregistration: {HOST.HOST}",
        "message": "Host name: {HOST.HOST}\r\nHost IP: {HOST.IP}\r\nAgent port: {HOST.PORT}"
    }
],
"parameters": []
},
{
    "mediatypeid": "3",
    "type": "2",
    "name": "SMS",
    "smtp_server": "",
    "smtp_helo": "",
    "smtp_email": "",
    "exec_path": "",
    "gsm_modem": "/dev/ttyS0",
    "username": "",
    "passwd": "",
    "status": "0",
    "smtp_port": "25",
    "smtp_security": "0",
    "smtp_verify_peer": "0",
    "smtp_verify_host": "0",
    "smtp_authentication": "0",
    "exec_params": "",
    "maxsessions": "1",
    "maxattempts": "3",
    "attempt_interval": "10s",
    "content_type": "1",
    "script": "",
    "timeout": "30s",
    "process_tags": "0",
    "show_event_menu": "1",
    "event_menu_url": "",
    "event_menu_name": "",
    "description": "",
    "message_templates": [
        {
            "eventsourc": "0",
            "recovery": "0",
            "subject": "",
            "message": "{EVENT.SEVERITY}: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT.UPDATE.TIME}",
        },
        {
            "eventsourc": "0",
            "recovery": "1",

```



```

        "subject": "",
        "message": "RESOLVED: {EVENT.NAME}\r\nHost: {HOST.NAME}\r\n{EVENT.DATE} {EVENT.TIME}"
    },
    {
        "eventsourc": "0",
        "recovery": "2",
        "subject": "",
        "message": "{USER.FULLNAME} {EVENT.UPDATE.ACTION} problem at {EVENT.UPDATE.DATE} {EVENT.UPDATE.TIME}"
    },
    {
        "eventsourc": "1",
        "recovery": "0",
        "subject": "",
        "message": "Discovery: {DISCOVERY.DEVICE.STATUS} {DISCOVERY.DEVICE.IPADDRESS}"
    },
    {
        "eventsourc": "2",
        "recovery": "0",
        "subject": "",
        "message": "Autoregistration: {HOST.HOST}\r\nHost IP: {HOST.IP}\r\nAgent port: {HOST.PORT}"
    }
],
"parameters": []
},
"id": 1
}

```

Veja também

- [User](#)

Fonte

CMediaType::get() in ui/include/classes/api/services/CMediaType.php.

mediatype.update

Descrição

object mediatype.update(object/array mediaTypes)

Este método permite atualizar os tipos de mídia existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Parâmetros

(object/array) Propriedades do tipo de mídia a serem atualizadas.

A propriedade mediatypeid deve ser definida para cada tipo de mídia, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades de tipo de mídia padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
parameters	array	Parâmetros do webhook para substituir os parâmetros atuais do webhook.
message_templates	array	Modelos de mensagem para substituir os modelos de mensagem atuais.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tipos de mídia atualizados na propriedade `mediatypeids`.

Exemplos

Habilitando um tipo de mídia

Habilite um tipo de mídia, ou seja, defina seu status como "0".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "mediatype.update",
  "params": {
    "mediatypeid": "6",
    "status": "0"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "mediatypeids": [
      "6"
    ]
  },
  "id": 1
}
```

Fonte

CMediaType::update() in ui/include/classes/api/services/CMediaType.php.

Token

Esta classe foi projetada para trabalhar com tokens.

Referências de objetos:

- [Token](#)

Métodos disponíveis:

- [token.create](#) - criar novos tokens
- [token.delete](#) - excluir tokens
- [token.get](#) - recuperar tokens
- [token.update](#) - atualizar tokens
- [token.generate](#) - gerar tokens

> Objeto token

Os objetos a seguir estão diretamente relacionados à API token.

Token

O objeto token tem as seguintes propriedades.

Propriedade	Type	Descrição
tokenid	string	(somente leitura) ID do token.
name (requerido)	string	Nome do token.
description	text	Descrição do token.

Propriedade	Type	Descrição
userid	string	(somente leitura para atualização) Um usuário ao qual o token foi atribuído.
lastaccess	timestamp	Padrão: usuário atual. (somente leitura) Data e hora mais recentes em que o token foi autenticado.
status	integer	Zero se o token nunca foi autenticado. Status do token.
expires_at	timestamp	Valores possíveis: 0 - (padrão) token ativado; 1 - token desativado. Data e hora de expiração do token.
created_at	timestamp	Zero para tokens que nunca expiram.
creator_userid	string	(somente leitura) Data e hora de criação do token. (somente leitura) O usuário criador do token.

token.create

Descrição

object token.create(object/array tokens)

Este método permite criar novos tokens.

Note:

Apenas o tipo de usuário Super admin tem permissão para gerenciar tokens para outros usuários.

Note:

Um token criado por este método deve ser **gerado** antes de ser usado.

Parâmetros

(objeto/array) Tokens para criar.

O método aceita tokens com as **propriedades de token padrão**.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tokens criados na propriedade tokenids. A ordem dos IDs retornados corresponde à ordem dos tokens passados.

Exemplos

Criar um token

Crie um token habilitado que nunca expire e autentique o usuário de ID 2.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.create",
  "params": {
    "name": "Your token",
    "userid": "2"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tokenids": [
      "188"
    ]
  },
  "id": 1
}
```

Create a disabled token that expires at January 21st, 2021. This token will authenticate current user.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.create",
  "params": {
    "name": "Your token",
    "status": "1",
    "expires_at": "1611238072"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tokenids": [
      "189"
    ]
  },
  "id": 1
}
```

Fonte

CToken::create() in ui/include/classes/api/services/CToken.php.

token.delete

Descrição

object token.delete(array tokenids)

Este método permite excluir tokens.

Note:

Apenas o tipo de usuário Super admin tem permissão para gerenciar tokens para outros usuários.

Parâmetros

(array) IDs dos tokens a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tokens excluídos na propriedade tokenids.

Exemplos

Delete multiple tokens

Delete two tokens.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.delete",
  "params": [
    "188",
    "192"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tokenids": [
      "188",
      "192"
    ]
  },
  "id": 1
}
```

Fonte

CToken::delete() in ui/include/classes/api/services/CToken.php.

token.generate

Descrição

object token.generate(array tokenids)

Este método permite gerar tokens.

Note:

Apenas o tipo de usuário Super admin tem permissão para gerenciar tokens para outros usuários.

Parâmetros

(array) IDs dos tokens a serem gerados.

Valores de retorno

(array) Retorna um array de objetos contendo o ID do token gerado na propriedade tokenId e a string de autorização gerada na propriedade token.

Propriedade	Tipo	Descrição
tokenId	string	ID do token.
token	string	A string de autorização gerada para este token.

Exemplos

Gerar vários tokens

Gerar dois tokens.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.generate",
  "params": [
    "1",
    "2"
  ]
}
```

```

    ],
    "auth": "3a57200802b24cda67c4e4010b50c065",
    "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "tokenid": "1",
      "token": "bbcfce79a2d95037502f7e9a534906d3466c9a1484beb6ea0f4e7be28e8b8ce2"
    },
    {
      "tokenid": "2",
      "token": "fa1258a83d518eabd87698a96bd7f07e5a6ae8aeb8463cae33d50b91dd21bd6d"
    }
  ],
  "id": 0
}

```

Fonte

CToken::generate() in ui/include/classes/api/services/CToken.php.

token.get

Descrição

integer/array token.get(object parameters)

O método permite recuperar tokens de acordo com os parâmetros fornecidos.

Note:

Apenas o tipo de usuário Super admin tem permissão para visualizar tokens para outros usuários.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Type	Descrição
tokenids	string/array	Retorna apenas tokens com os IDs fornecidos.
userids	string/array	Retorna apenas os tokens criados para os usuários fornecidos.
token	string	Retorna apenas os tokens criados para o token de autenticação fornecido.
valid_at	timestamp	Retorna apenas tokens válidos (não expirados) na data e hora especificadas.
expired_at	timestamp	Retorna apenas os tokens que expiraram (não são válidos) na data e hora especificadas.
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: tokenid, name, lastaccess, status, expires_at e created_at. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	

Parâmetro	Type	Descrição
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperar um token

Recupere todos os dados do token com ID "2".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.get",
  "params": {
    "output": "extend",
    "tokenids": "2"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "tokenid": "1",
      "name": "The Token",
      "description": "",
      "userid": "1",
      "lastaccess": "0",
      "status": "0",
      "expires_at": "1609406220",
      "created_at": "1611239454",
      "creator_userid": "1"
    }
  ],
  "id": 1
}
```

Fonte

CToken::get() in ui/include/classes/api/services/CToken.php.

token.update

Descrição

object token.update(object/array tokens)

Este método permite atualizar os tokens existentes.

Note:

Apenas o tipo de usuário Super admin tem permissão para gerenciar tokens para outros usuários.

Parâmetros

(object/array) Propriedades do token a serem atualizadas.

A propriedade `tokenid` deve ser definida para cada token, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

O método aceita tokens com as **propriedades de token padrão**.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos tokens atualizados na propriedade `tokenids`.

Exemplos

Remover expiração do token

Remover data de expiração do token.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "token.update",
  "params": {
    "tokenid": "2",
    "expires_at": "0"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "tokenids": [
      "2"
    ]
  },
  "id": 1
}
```

Fonte

`CToken::update()` in `ui/include/classes/api/services/CToken.php`.

Trigger

Esta classe foi projetada para trabalhar com trigger.

Referências de objetos:

- **Trigger**

Available methods:

- **trigger.adddependencies** - adicionando novas dependências de trigger
- **trigger.create** - criando novas triggers
- **trigger.delete** - excluindo triggers
- **trigger.deletedependencies** - excluindo dependências de trigger
- **trigger.get** - recuperando triggers
- **trigger.update** - atualizando triggers

> Objeto de trigger

Os objetos a seguir estão diretamente relacionados à API `trigger`.

Trigger

The trigger object has the following properties.

Property	Type	Description
triggerid	string	(readonly) ID of the trigger.
description (required)	string	Name of the trigger.
expression (required)	string	Reduced trigger expression.
event_name	string	Event name generated by the trigger.
opdata	string	Operational data.
comments	string	Additional description of the trigger.
error	string	(readonly) Error text if there have been any problems when updating the state of the trigger.
flags	integer	(readonly) Origin of the trigger. Possible values are: 0 - (default) a plain trigger; 4 - a discovered trigger.
lastchange	timestamp	(readonly) Time when the trigger last changed its state.
priority	integer	Severity of the trigger. Possible values are: 0 - (default) not classified; 1 - information; 2 - warning; 3 - average; 4 - high; 5 - disaster.
state	integer	(readonly) State of the trigger. Possible values: 0 - (default) trigger state is up to date; 1 - current trigger state is unknown.

Property	Type	Description
status	integer	Whether the trigger is enabled or disabled. Possible values are: 0 - (default) enabled; 1 - disabled.
templateid	string	(readonly) ID of the parent template trigger.
type	integer	Whether the trigger can generate multiple problem events. Possible values are: 0 - (default) do not generate multiple events; 1 - generate multiple events.
url	string	URL associated with the trigger.
value	integer	(readonly) Whether the trigger is in OK or problem state. Possible values are: 0 - (default) OK; 1 - problem.
recovery_mode	integer	OK event generation mode. Possible values are: 0 - (default) Expression; 1 - Recovery expression; 2 - None.
recovery_expression	string	Reduced trigger recovery expression.
correlation_mode	integer	OK event closes. Possible values are: 0 - (default) All problems; 1 - All problems if tag values match.
correlation_tag	string	Tag for matching.

Property	Type	Description
manual_close	integer	Allow manual close.
uuid	string	<p>Possible values are: 0 - (default) No; 1 - Yes.</p> <p>Universal unique identifier, used for linking imported triggers to already existing ones. Used only for triggers on templates. Auto-generated, if not given.</p> <p>For update operations this field is readonly.</p>

Etiqueta de trigger

O objeto da etiqueta de trigger tem as seguintes propriedades..

Propriedade	Type	Descrição
tag (requerido)	string	Nome da etiqueta de trigger.
value	string	Valor da tag etiqueta de trigger.

trigger.adddependencies

Descrição

`object trigger.adddependencies(object/array triggerDependencies)`

Este método permite criar novas dependências de trigger.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object/array) Dispara dependências para criar.

Cada dependência de trigger tem os seguintes parâmetros:

Parâmetro	Type	Descrição
triggerid (requerido)	string	ID da trigger dependente.
dependsOnTriggerid (requerido)	string	ID da trigger do qual a trigger depende.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos gatilhos dependentes na propriedade `triggerids`.

Exemplos

Adicionar uma dependência de trigger

Torne o trigger "14092" dependente do trigger "13565."

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.adddependencies",
  "params": {
    "triggerid": "14092",
    "dependsOnTriggerid": "13565"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "14092"
    ]
  },
  "id": 1
}
```

Veja também

- [trigger.update](#)
- [Dependências de trigger](#)

Fonte

CTrigger::addDependencies() in ui/include/classes/api/services/CTrigger.php.

trigger.create

Descrição

`object trigger.create(object/array triggers)`

Este método permite criar novas triggers.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário].

Parâmetros

(objeto/array) Triggers para criar.

Além das [propriedades padrão do gatilho](#), o método aceita os seguintes parâmetros.

Parâmetro	Type	Descrição
dependencies	array	Triggers dos quais o trigger é dependente.
tags	array	As triggers devem ter a propriedade triggerid definida. Trigger tags .

Attention:

A expressão de trigger deve ser fornecida em sua forma expandida.

Valores de retorno

(object) Retorna um objeto contendo os IDs das triggers criados na propriedade `triggerids`. A ordem dos IDs retornados corresponde à ordem das triggers passadas.

Exemplos

Criando uma trigger

Criar uma trigger com uma única dependência de trigger.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.create",
  "params": [
    {
      "description": "Processor load is too high on {HOST.NAME}",
      "expression": "last(/Linux server/system.cpu.load[percpu,avg1])>5",
      "dependencies": [
        {
          "triggerid": "17367"
        }
      ]
    },
    {
      "description": "Service status",
      "expression": "length(last(/Linux server/log[/var/log/system,Service .* has stopped]))<>0",
      "dependencies": [
        {
          "triggerid": "17368"
        }
      ],
      "tags": [
        {
          "tag": "service",
          "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\\\1\")}"
        },
        {
          "tag": "error",
          "value": ""
        }
      ]
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "17369",
      "17370"
    ]
  },
  "id": 1
}
```

Fonte

CTrigger::create() in ui/include/classes/api/services/CTrigger.php.

trigger.delete

Descrição

`object trigger.delete(array triggerIds)`

Este método permite deletar triggers.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs das trigger a serem excluídas.

Valores de retorno

(object) Retorna um objeto contendo os IDs das triggers excluídas na propriedade `triggerids`.

Exemplos

Excluir várias triggers

Excluir duas triggers.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.delete",
  "params": [
    "12002",
    "12003"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "12002",
      "12003"
    ]
  },
  "id": 1
}
```

Fonte

`CTrigger::delete()` in `ui/include/classes/api/services/CTrigger.php`.

trigger.deletedependencies

Descrição

`object trigger.deletedependencies(string/array triggers)`

Este método permite excluir todas as dependências de trigger a partir dos triggers fornecidos.

Note:

Esse método está disponível apenas para usuários Admin e Super admin. Permissões para chamar o método pode ser revogada nas configurações de função do usuário. Para mais informações, consulte [User roles](#)

Parâmetros

Excluir triggers (string/array) a partir das dependências de trigger.

Valores retornados

Retorna um objeto (object) contendo os IDs dos triggers afetados sob a propriedade `triggerids`.

Exemplos

Deletando dependências de vários triggers

Delete all dependencies from two triggers.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.deleteDependencies",
  "params": [
    {
      "triggerid": "14544"
    },
    {
      "triggerid": "14545"
    }
  ],
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "14544",
      "14545"
    ]
  },
  "id": 1
}
```

Ver também

- [trigger.update](#)

Fonte

CTrigger::deleteDependencies() in ui/include/classes/api/services/CTrigger.php.

trigger.get

Descrição

integer/array trigger.get(object parameters)

O método permite recuperar triggers de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
triggerids	string/array	Retornar apenas triggers com os IDs fornecidos.

Parâmetro	Tipo	Descrição
groupids	string/array	Retornar apenas triggers que pertencem a hosts dos grupos de hosts fornecidos.
templateids	string/array	Retornar apenas triggers que pertencem aos modelos fornecidos.
hostids	string/array	Retorna apenas as triggers que pertencem aos hosts fornecidos.
itemids	string/array	Retorna apenas as triggers que contêm os itens fornecidos.
functions	string/array	Retorne apenas triggers que usam as funções fornecidas.
		Consulte a página de função compatível para obter uma lista de funções compatíveis.
group	string	Retorna apenas as triggers que pertencem a hosts do grupo de hosts com o nome fornecido.
host	string	Retorna apenas as triggers que pertencem ao host com o nome fornecido.
inherited	boolean	Se definido como true, retorna apenas triggers herdadas de um modelo.
templated	boolean	Se definido como true, retornará apenas as triggers que pertencem aos modelos.
dependent	boolean	Se definido como true, retorna apenas as triggers que possuem dependências. Se definido como false, retorne apenas as triggers que não possuem dependências.
monitored	flag	Retorna apenas triggers habilitadas que pertencem a hosts monitorados e contêm apenas itens habilitados.
active	flag	Retorna apenas triggers habilitadas que pertencem a hosts monitorados.
manutenção	boolean	Se definido como true, retornará apenas as triggers habilitadas que pertencem aos hosts em manutenção.
withUnacknowledgedEvents	flag	Retorna apenas triggers que possuem eventos não reconhecidos.
withAcknowledgedEvents	flag	Retorna apenas triggers com todos os eventos reconhecidos.
withLastEventUnacknowledged	flag	Retorna somente triggers com o último evento não reconhecido.
skipDependent	flag	Ignorar triggers em um estado de problema que são dependentes de outras triggers. Observe que as outras triggers são ignoradas se estiverem desabilitadas, tiverem itens desabilitados ou hosts de itens desabilitados.
lastChangeSince	timestamp	Retorna apenas as triggers que mudaram de estado após o tempo determinado.
lastChangeTill	timestamp	Retorna apenas as triggers que mudaram seu estado antes do tempo determinado.
only_true	flag	Retorna apenas triggers que estiveram recentemente em um estado de problema.
min_severity	integer	Retorna apenas triggers com gravidade maior ou igual à gravidade especificada.
evaltype	integer	Regras para pesquisa de etiquetas.
		Valores possíveis: 0 - (padrão) E/ou; 2 - Ou.
tags	array of objects	Retorna apenas triggers com as etiquetas fornecidas. Correspondência exata por etiquetas e pesquisa com distinção entre maiúsculas e minúsculas por valor de etiquetas, dependendo do valor do operador. Formato: [{"tag": "<tag>", "value": "<value>", "operator": "<operator>"}, ...]. Um array vazio retorna todos as triggers.
		Possíveis tipos de operadores: 0 - (padrão) Like; 1 - Equal; 2 - Not like; 3 - Not equal 4 - Exists; 5 - Not exists.
expandComment	flag	Expandir macros na descrição da trigger .

Parâmetro	Tipo	Descrição
expandDescription	flag	Expandir macros no nome da trigger.
expandExpression	flag	Expandir funções e macros na expressão da trigger.
selectGroups	query	Retorne os grupos de hosts aos quais a trigger pertence na propriedade groups .
selectHosts	query	Retorne os hosts aos quais a trigger pertence na propriedade hosts .
selectItems	query	Retorna os itens contidos pelo trigger na propriedade items .
selectFunctions	query	Retorne funções usadas na trigger na propriedade functions .
		Os objetos de função representam as funções usadas na expressão de trigger e possuem as seguintes propriedades: functionid - (string) ID da função; itemid - (string) ID do item usado na função; function - (string) nome da função; parameter - (string) parâmetro passado para a função. O parâmetro de consulta é substituído pelo símbolo \$ na string retornada.
selectDependencies	query	Retorne triggers dos quais a trigger depende na propriedade dependencies .
selectDiscoveryRule	query	Retorne a regra de descoberta de baixo nível que criou a trigger.
selectLastEvent	query	Retorne o último evento de disparo significativo na propriedade lastEvent .
selectTags	query	Retorne as etiquetas de trigger na propriedade etiquetas .
selectTriggerDiscovery	query	Retorne o objeto de descoberta de trigger na propriedade triggerDiscovery . Os objetos de descoberta de trigger vinculam a trigger a um protótipo de trigger a partir do qual ele foi criado.
		Ele tem as seguintes propriedades: parent_triggerid - (string) ID do protótipo do gatilho a partir do qual a trigger foi criada.
filter	object	Retorna apenas os resultados que correspondem exatamente ao filtro fornecido.
		Aceita uma array, em que as chaves são nomes de propriedades e os valores são um único valor ou uma array de valores para correspondência.
		Suporta filtros adicionais: host - nome técnico do host ao qual a trigger pertence; hostid - ID do host ao qual a trigger pertence.
limitSelects	integer	Limite o número de registros retornados por subseleções.
		Aplica-se às seguintes subseleções: selectHosts - os resultados serão classificados por host . Classificar o resultado pelas propriedades fornecidas.
sortfield	string/array	
		Os valores possíveis são: triggerid , description , status , prioridade , lastchange e hostname .
countOutput	boolean	Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes na página comentário de referência .
editable	boolean	
excludeSearch	boolean	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando dados por ID de trigger

Recupere todos os dados e as funções usadas na trigger "14062".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.get",
  "params": {
    "triggerids": "14062",
    "output": "extend",
    "selectFunctions": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "triggerid": "14062",
      "expression": "{13513}<10m",
      "description": "{HOST.NAME} has been restarted (uptime < 10m)",
      "url": "",
      "status": "0",
      "value": "0",
      "priority": "2",
      "lastchange": "0",
      "comments": "The host uptime is less than 10 minutes",
      "error": "",
      "templateid": "10016",
      "type": "0",
      "state": "0",
      "flags": "0",
      "recovery_mode": "0",
      "recovery_expression": "",
      "correlation_mode": "0",
      "correlation_tag": "",
      "manual_close": "0",
      "opdata": "",
      "functions": [
        {
          "functionid": "13513",
          "itemid": "24350",
          "triggerid": "14062",
          "parameter": "$",
          "function": "last"
        }
      ]
    }
  ],
  "id": 1
}
```

Recuperando triggers em estado de problema

Recupere o ID, o nome e a gravidade de todos as triggers no estado do problema e classifique-as por gravidade em ordem decrescente.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.get",
  "params": {
    "output": [
      "triggerid",
      "description",
      "priority"
    ],
    "filter": {
      "value": 1
    },
    "sortfield": "priority",
    "sortorder": "DESC"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "triggerid": "13907",
      "description": "Zabbix self-monitoring processes < 100% busy",
      "priority": "4"
    },
    {
      "triggerid": "13824",
      "description": "Zabbix discoverer processes more than 75% busy",
      "priority": "3"
    }
  ],
  "id": 1
}
```

Recuperando uma trigger específica com etiquetas

Recuperar uma trigger específica com etiquetas.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.get",
  "params": {
    "output": [
      "triggerid",
      "description"
    ],
    "selectTags": "extend",
    "triggerids": [
      "17578"
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "triggerid": "17370",
      "description": "Service status",
      "tags": [
        {
          "tag": "service",
          "value": "{{ITEM.VALUE}.regsub(\"Service (.*) has stopped\", \"\\1\")}"
        },
        {
          "tag": "error",
          "value": ""
        }
      ]
    }
  ],
  "id": 1
}

```

Veja também

- [discovery rule](#)
- [Item](#)
- [Host](#)
- [Host group](#)

Fonte

CTrigger::get() in ui/include/classes/api/services/CTrigger.php.

trigger.update

Descrição

object trigger.update(object/array triggers)

Este método permite atualizar as triggers existentes.

Note:

Este método está disponível apenas para os tipos de usuário Admin e Super admin. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [User roles](#) para obter mais informações.

Parâmetros

(object/array) Propriedades da trigger a serem atualizadas.

A propriedade `triggerid` deve ser definida para cada trigger, todas as outras propriedades são opcionais. Apenas as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Além das [propriedades padrão da trigger](#), o método aceita os seguintes parâmetros.

Parâmetro	Type	Descrição
dependencies	array	Acionadores dos quais a trigger depende.
tags	array	As trigger devem ter a propriedade <code>triggerid</code> definida. Trigger tags .

Attention:

A expressão de trigger deve ser fornecida em sua forma expandida.

Valores de retorno

(object) Retorna um objeto contendo os IDs das triggers atualizados na propriedade `triggerids`.

Exemplos

Habilitando uma trigger

Habilite uma trigger, ou seja, defina seu status para 0.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.update",
  "params": {
    "triggerid": "13938",
    "status": 0
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "13938"
    ]
  },
  "id": 1
}
```

Substituindo etiquetas de triggers

Substituir etiqueta para trigger.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "trigger.update",
  "params": {
    "triggerid": "13938",
    "tags": [
      {
        "tag": "service",
        "value": "{{ITEM.VALUE}}.regsub(\"Service (.*) has stopped\", \"\\1\")"
      },
      {
        "tag": "error",
        "value": ""
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "triggerids": [
      "13938"
    ]
  },
  "id": 1
}
```

```
}    "id": 1
```

Veja também

- [trigger.adddependencies](#)
- [trigger.deletedependencies](#)

Fonte

CTrigger::update() in ui/include/classes/api/services/CTrigger.php.

Usuário

Esta classe é planejada para trabalhar com usuários.

Referências do objeto:

- [Usuário](#)

Métodos disponíveis:

- [user.checkauthentication](#) - verificando e prolongando as sessões do usuário
- [user.create](#) - criar novos usuários
- [user.delete](#) - deletar usuários
- [user.get](#) - recuperar usuários
- [user.login](#) - fazer login na API
- [user.logout](#) - fazer logout na API
- [user.unblock](#) - desbloquear usuários
- [user.update](#) - atualizar usuários

>Usuário object

Os objetos a seguir estão diretamente relacionados aouser API.

Usuário

O objeto do usuário tem as seguintes propriedades:

Propriedade	Tipo	Descrição
userid	string	(aleatoriamente) ID do usuário.
username (requisitado)	string	Nome do usuário.
roleid (requisitado)	string	A ID do papel do usuário.
attempt_clock	timestamp	(aleatoriamente) Horário da última tentativa de login.
attempt_failed	integer	(aleatoriamente) Número de tentativa de login sem sucesso recente.
attempt_ip	string	(aleatoriamente) Endereço do IP da última tentativa de login sem sucesso.
autologin	integer	Se permitir o auto-login. Possível valores: 0 - (padrão) auto-login desabilitado; 1 - auto-login habilitado.
autologout	string	Tempo de sessão do usuário. Aceita segundos e unidade de tempo com sufixos seconds and time. Se for 0s, a sessão nunca irá expirar.
lang	string	Padrão: 15m. Código da linguagem do usuário, por exemplo, en_GB.
name	string	Padrão: default - padrão do sistema. Nome do usuário.

Propriedade	Tipo	Descrição
refresh	string	Período de atualização automática. Aceita segundos e unidade de tempo com sufixos (e.g., 30s, 90s, 1m, 1h).
rows_per_page	integer	Padrão: 30s. Quantidade de linhas por página.
surname	string	Padrão: 50. Sobrenome do usuário.
theme	string	Tema do usuário.
url	string	Valores possíveis: default - (padrão) padrão do sistema; blue-theme - Azul; dark-theme - Escuro. Redirecionamento da URL da página depois do usuário fazer log in.
timezone	string	Fuso horário do usuário, por exemplo, Europe/London, UTC. Padrão: default - padrão do sistema.
alias (descontinuado)	string	Para a lista completa dos fusos horários por favor recorrer a PHP documentation . Esta propriedade é descontinuada, por favor use username ao invés de. Usuário alias.

Note que para alguns métodos (update, delete) a combinação do parâmetro requisitado/opcional é diferente. unsuccessful login attempt.| [attempt_failed|integer|(readonly) Recent failed login attempt count.| [attempt_ip|string|(readonly) IP address from where the last unsuccessful login attempt came from.| [autologin|integer|Whether to enable auto-login.

Possible values:
0 - (default) auto-login disabled;
1 - auto-login enabled.| [autologout|string|User session life time. Accepts seconds and time unit with suffix. If set to 0s, the session will never expire.

Default: 15m.| [lang|string|Language code of the user's language, for example, en_GB.

Default: default - system default.| [name|string|Name of the user.| [refresh|string|Automatic refresh period. Accepts seconds or time unit with suffix (e.g., 30s, 90s, 1m, 1h).

Default: 30s.| [rows_per_page|integer|Amount of object rows to show per page.

Default: 50.| [surname|string|Surname of the user.| [theme|string|User's theme.

Possible values:
default - (default) system default;
blue-theme - Blue;
dark-theme - Dark.| [url|string|URL of the page to redirect the user to after logging in.| [timezone|string|User's time zone, for example, Europe/London, UTC.

Default: default - system default.

For the full list of supported time zones please refer to [PHP documentation](#).| [alias
(deprecated)|string|This property is deprecated, please use username instead.
User alias.]

Note that for some methods (update, delete) the required/optional parameter combination is different.

Mídia

O objeto mídia tem as seguintes propriedades:

Propriedade	Tipo	Descrição
mediatypeid (required)	string	ID do tipo de mídia usado pela mídia.
sendto (required)	string/array	Endereço, nome do usuário ou outro identificador do receptor. Se o tipo Media type for e-mail, valores serão apresentados como array. Por outros tipos de Media types , o valor é apresentado como uma string.
active	integer	Se a mídia estiver habilitada. Valores possíveis: 0 - (padrão) habilitado; 1 - desabilitado.

Propriedade	Tipo	Descrição
severity	integer	<p>Severidades de Trigger para enviar notificações sobre.</p> <p>Severidade são armazenadas de forma binária, com cada bit representando a severidade correspondente. Por exemplo, 12 em binário é 1100, e isso significa que as notificações serão enviadas a partir de triggers com severidade de aviso e média.</p> <p>Consultar trigger object page para obter uma lista de severidades de trigger suportadas.</p>
period	string	<p>Padrão: 63</p> <p>Horário quando as notificações podem ser enviadas como um time period ou macros do usuário separados por ponto e vírgula.</p> <p>Padrão: 1-7,00:00-24:00</p>

user.checkAuthentication

Descrição

objeto `user.checkAuthentication`

Este método verifica e prolonga a sessão do usuário.

::: não importante

Chamar o método **user.checkAuthentication** prolonga a sessão do usuário por padrão. :::

Parâmetros

O método aceita os seguintes parâmetros.

Parâmetro	Type	Descrição
extend	boolean	Valor padrão: "true". Definir seu valor como "false" permite verificar a sessão sem estender sua vida útil. Suportado desde o Zabbix 4.0.
sessionid	string	ID da sessão do usuário.

Valores de retorno

(object) Retorna um objeto contendo informações sobre o usuário.

Exemplos

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.checkAuthentication",
  "params": {
    "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
  },
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userid": "1",
    "username": "Admin",
    "name": "Zabbix",
    "surname": "Administrator",
  }
}
```



```

        "url": "",
        "autologin": "1",
        "autologout": "0",
        "lang": "ru_RU",
        "refresh": "0",
        "theme": "default",
        "attempt_failed": "0",
        "attempt_ip": "127.0.0.1",
        "attempt_clock": "1355919038",
        "rows_per_page": "50",
        "timezone": "Europe/Riga",
        "roleid": "3",
        "type": 3,
        "sessionid": "673b8ba11562a35da902c66cf5c23fa2"
        "debug_mode": 0,
        "userip": "127.0.0.1",
        "gui_access": 0
    },
    "id": 1
}

```

Note:

Response is similar to [User.login](#) call response with "userData" parameter set to true (the difference is that user data is retrieved by session id and not by username / password).

Fonte

CUser::checkAuthentication() in ui/include/classes/api/services/CUser.php.

user.create

Descrição

object user.create(usuários de objeto/array)

Este método permite criar novos usuários.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções de usuário](#) para obter mais informações.

Note:

A força/qualidade da senha do usuário é validada de acordo com as regras de política de senha definidas pela API de autenticação. Consulte [API de autenticação](#) para obter mais informações.

Parâmetros

(objeto/array) Usuários a serem criados.

Além das [propriedades de usuário padrão](#), o método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
passwd (requerido)	string	Senha do usuário. Pode ser omitido se o usuário for adicionado apenas a grupos que tenham acesso LDAP.
usrgrps (requerido)	array	Usuário grupos ao qual adicionar o usuário.
medias	array	Os grupos de usuários devem ter a propriedade usrgrpId definida. Usuário mídia a ser criado.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos usuários criados na propriedade `userids`. A ordem dos IDs retornados corresponde à ordem dos usuários passados.

Exemplos

Criando um usuário

Crie um novo usuário, adicione-o a um grupo de usuários e crie uma nova mídia para ele.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.create",
  "params": {
    "username": "John",
    "passwd": "Doe123",
    "roleid": "5",
    "usrgrps": [
      {
        "usrgrpid": "7"
      }
    ],
    "medias": [
      {
        "mediatypeid": "1",
        "sendto": [
          "support@company.com"
        ],
        "active": 0,
        "severity": 63,
        "period": "1-7,00:00-24:00"
      }
    ]
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userids": [
      "12"
    ]
  },
  "id": 1
}
```

Veja também

- [Autenticação](#)
- [Mídia](#)
- [Grupo de usuários](#)
- [Função](#)

Fonte

CUser::create() in ui/include/classes/api/services/CUser.php.

user.delete

Descrição

object user.delete(array users)

Este método permite excluir usuários.

Note:

Este método está disponível apenas para o tipo de usuário Super admin.

As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) IDs de usuários a serem excluídos.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos usuários excluídos na propriedade `userids`.

Exemplos

Excluindo vários usuários

Excluir dois usuários.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.delete",
  "params": [
    "1",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userids": [
      "1",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

`CUser::delete()` in `ui/include/classes/api/services/CUser.php`.

user.get

Descrição

`integer/array user.get(object parameters)`

O método permite recuperar usuários de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário (uer roles). Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos parâmetros a seguir.

Parâmetro	Tipo	Descrição
mediaids	string/array	Retorna apenas usuários que usam a mídia fornecida.
mediatypesids	string/array	Retorna apenas usuários que usam os tipos de mídia fornecidos.
userids	string/array	Retorne apenas usuários com os IDs fornecidos.
usrgrpids	string/array	Retorna apenas usuários que pertencem aos grupos de usuários especificados.
getAccess	flag	Adiciona informações adicionais sobre as permissões do usuário. Adiciona as seguintes propriedades para cada usuário: gui_access - (integer) método de autenticação de frontend do usuário. Consulte a propriedade gui_access do objeto de grupo de usuários para obter uma lista de valores possíveis. debug_mode - (integer) indica se debug está habilitado para o usuário. Valores possíveis: 0 - depuração desabilitada, 1 - depuração habilitada. users_status - (integer) indica se o usuário está desabilitado. Valores possíveis: 0 - usuário habilitado, 1 - usuário desabilitado.
selectMedias	query	Retorna a mídia usada pelo usuário na propriedade medias .
selectMediatypes	query	Retorna os tipos de mídia usados pelo usuário na propriedade mediatypes .
selectUsrgrps	query	Retorna grupos de usuários aos quais o usuário pertence na propriedade usrgrps .
selectRole	query	Retorne a função do usuário na propriedade role .
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas.
countOutput	boolean	Os valores possíveis são: userid e username. Esses parâmetros sendo comuns para todos os métodos get são descritos em detalhes no comentário de referência .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Valores de retorno

(integer/array) Retorna:

- uma array de objetos;
- a contagem de objetos recuperados, se o parâmetro countOutput foi usado.

Exemplos

Recuperando usuários

Recupere todos os usuários configurados.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.get",
  "params": {
    "output": "extend"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "userid": "1",
      "username": "Admin",
      "name": "Zabbix",
      "surname": "Administrator",
      "url": "",
      "autologin": "1",
      "autologout": "0",
      "lang": "en_GB",
      "refresh": "0s",
      "theme": "default",
      "attempt_failed": "0",
      "attempt_ip": "",
      "attempt_clock": "0",
      "rows_per_page": "50",
      "timezone": "default",
      "roleid": "3"
    },
    {
      "userid": "2",
      "username": "guest",
      "name": "",
      "surname": "",
      "url": "",
      "autologin": "0",
      "autologout": "15m",
      "lang": "default",
      "refresh": "30s",
      "theme": "default",
      "attempt_failed": "0",
      "attempt_ip": "",
      "attempt_clock": "0",
      "rows_per_page": "50",
      "timezone": "default",
      "roleid": "4"
    },
    {
      "userid": "3",
      "username": "user",
      "name": "Zabbix",
      "surname": "User",
      "url": "",
      "autologin": "0",
      "autologout": "0",
      "lang": "ru_RU",
      "refresh": "15s",
      "theme": "dark-theme",
      "attempt_failed": "0",
      "attempt_ip": "",
      "attempt_clock": "0",
      "rows_per_page": "100",
      "timezone": "default",
      "roleid": "1"
    }
  ],
  "id": 1
}

```

Recuperando dados do usuário

Recuperar dados de um usuário com ID "12". Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.get",
  "params": {
    "output": ["userid", "username"],
    "selectRole": "extend",
    "userids": "12"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "userid": "12",
      "username": "John",
      "role": {
        "roleid": "5",
        "name": "Operator",
        "type": "1",
        "readonly": "0"
      }
    }
  ],
  "id": 1
}
```

Veja também

- [Mídia](#)
- [Tipo de mídia](#)
- [Grupo de usuários](#)
- [Função](#)

Fonte

CUser::get() in ui/include/classes/api/services/CUser.php.

user.login

Descrição

string/object user.login(object parameters)

Este método permite efetuar login na API e gerar um token de autenticação.

Ao usar este método, você também precisa fazer **user.logout** para evitar a geração de um grande número de registros de sessão aberta.

::: não importante

Este método está disponível apenas para usuários não autenticados e deve ser chamado sem o parâmetro `auth` na solicitação JSON-RPC. :::

Parâmetros

(object) Parâmetros contendo o nome de usuário e senha.

O método aceita os seguintes parâmetros.

Parâmetro	Tipo	Descrição
password (requerido)	string	Senha do usuário.

Parâmetro	Tipo	Descrição
username (requerido)	string	Nome de usuário.
userData	flag	Retorna informações sobre o usuário autenticado.

Valores de retorno

(string/object) Se o parâmetro userData for usado, retorna um objeto contendo informações sobre o usuário autenticado.

Além das **propriedades de usuário padrão**, as seguintes informações são retornadas:

Propriedade	Tipo	Descrição
debug_mode	boolean	Se o modo de depuração está ativado para o usuário.
gui_access	integer	Método de autenticação do usuário para o frontend.
sessionid	string	Consulte a propriedade gui_access do user group object para uma lista de valores possíveis. Token de autenticação, que deve ser usado nas seguintes solicitações de API.
userip	string	endereço IP do usuário.

Note:

Se um usuário foi autenticado com sucesso após uma ou mais tentativas com falha, o método retornará os valores atuais para as propriedades attempt_clock, attempt_failed e attempt_ip e, em seguida, redefina-as.

Se o parâmetro userData não for usado, o método retornará um token de autenticação.

Note:

O token de autenticação gerado deve ser lembrado e usado no parâmetro auth das seguintes solicitações JSON-RPC. Também é necessário ao usar a autenticação HTTP.

Exemplos

Autenticação de um usuário

Autenticar um usuário.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.login",
  "params": {
    "username": "Admin",
    "password": "zabbix"
  },
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": "0424bd59b807674191e7d77572075f33",
  "id": 1
}
```

Solicitando informações do usuário autenticado

Autenticar e retornar informações adicionais sobre o usuário.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.login",
  "params": {
    "username": "Admin",
    "password": "zabbix",
    "userData": true
  },
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userid": "1",
    "username": "Admin",
    "name": "Zabbix",
    "surname": "Administrator",
    "url": "",
    "autologin": "1",
    "autologout": "0",
    "lang": "ru_RU",
    "refresh": "0",
    "theme": "default",
    "attempt_failed": "0",
    "attempt_ip": "127.0.0.1",
    "attempt_clock": "1355919038",
    "rows_per_page": "50",
    "timezone": "Europe/Riga",
    "roleid": "3",
    "type": 3,
    "debug_mode": 0,
    "userip": "127.0.0.1",
    "gui_access": "0",
    "sessionid": "5b56eee8be445e98f0bd42b435736e42"
  },
  "id": 1
}
```

Veja também

- [user.logout](#)

Fonte

CUser::login() in ui/include/classes/api/services/CUser.php.

user.logout

Descrição

string/objeto user.logout(array)

Este método permite sair da API e invalidar o token de autenticação atual.

Note:

Este método está disponível para usuários de qualquer tipo. As permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Consulte [Funções do usuário](#) para obter mais informações.

Parâmetros

(array) O método aceita um array vazio.

Valores de retorno

(boolean) Retorna true se o usuário foi desconectado com sucesso.

Exemplos

Log out

Log out da API.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.logout",
  "params": [],
  "id": 1,
  "auth": "16a46baf181ef9602e1687f3110abf8a"
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": true,
  "id": 1
}
```

Veja também

- [user.login](#)

Fonte

CUser::login() in ui/include/classes/api/services/CUser.php.

user.unblock

Descrição

object user.unblock(array userids)

Este método permite desbloquear usuários. This method allows to unblock users.

Note:

Esse método está disponível apenas para o usuário Super admin. Permissões para chamar o método pode ser revogada nas configurações de função do usuário. Para mais informações, consultar [User roles](#).

Parâmetros

IDs de usuários para desbloquear (array)

Valores de retorno

Retorna um objeto (object) contendo os IDs dos usuários desbloqueados sob a propriedade userids .

Exemplos

Desbloqueando vários usuários

Desbloquear dois usuários.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "user.unblock",
  "params": [
    "1",
    "5"
  ],
  "auth": "3a57200802b24cda67c4e4010b50c065",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userids": [
      "1",
      "5"
    ]
  },
  "id": 1
}
```

Fonte

CUser::unblock() in ui/include/classes/api/services/CUser.php.

user.update

Descrição

object user.update(object/array users)

Esse método permite atualizar usuários existentes.

Note:

Esse método está disponível para usuários de qualquer tipo. Permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Ver [User roles](#) para mais informações.

Note:

A força da senha do usuário é validada de acordo com as regras da política de senhas definidas pela API de Autenticação. Ver [Authentication API](#) para mais informações.

Parameters

(object/array) User properties to be updated.

The `userid` property must be defined for each user, all other properties are optional. Only the passed properties will be updated, all others will remain unchanged.

Additionally to the [standard user properties](#), the method accepts the following parameters.

Parameter	Type	Description
passwd	string	User's password. Can be empty string if user belongs to or is moved only to groups that have LDAP access.
usrgrps	array	User groups to replace existing user groups. The user groups must have the <code>usrgrpid</code> property defined.
medias	array	User media to replace existing media.

Return values

(object) Returns an object containing the IDs of the updated users under the `userids` property.

Exemplos

Renaming a user

Rename a user to John Doe.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.update",
  "params": {
    "userid": "1",
    "name": "John",
    "surname": "Doe"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userids": [
      "1"
    ]
  },
  "id": 1
}
```

Changing user role

Change a role of a user.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "user.update",
  "params": {
    "userid": "12",
    "roleid": "6"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "userids": [
      "12"
    ]
  },
  "id": 1
}
```

See also

- [Authentication](#)

Fonte

CUser::update() in ui/include/classes/api/services/CUser.php.

Value map

Esta classe é planejada para trabalhar com value maps.

Referências do objeto:

- [Value map](#)

Métodos disponíveis:

- [valuemap.create](#) - criar value map
- [valuemap.delete](#) - deletar value map
- [valuemap.get](#) - recuperar value map
- [valuemap.update](#) - atualizar value map

> Objeto de mapa de valor

Os objetos a seguir estão diretamente relacionados à `valuemap` API.

Value map

The value map object has the following properties.

Property	Type	Description
<code>valuemapid</code>	string	(readonly) ID of the value map.
hostid (required)	id	Value map host ID.
name (required)	string	Name of the value map.
mappings (required)	array	Value mappings for current value map. The mapping object is described in detail below .
<code>uuid</code>	string	Universal unique identifier, used for linking imported value maps to already existing ones. Used only for value maps on templates. Auto-generated, if not given. For update operations this field is readonly.

Value mappings

The value mappings object defines value mappings of the value map. It has the following properties.

Property	Type	Description
type	integer	Mapping match type. For type equal 0,1,2,3,4 value field cannot be empty, for type 5 value field should be empty. Possible values: 0 - (default) exact match ; 1 - mapping will be applied if value is greater or equal ¹ ; 2 - mapping will be applied if value is less or equal ¹ ; 3 - mapping will be applied if value is in range (ranges are inclusive), allow to define multiple ranges separated by comma character ¹ ; 4 - mapping will be applied if value match regular expression ² ; 5 - default value, mapping will be applied if no other match were found. Original value.
value (required)	string	Is not required for mapping of type "default".
newvalue (required)	string	Value to which the original value is mapped to.

¹ supported only for items having value type "numeric unsigned", "numeric float".

² supported only for items having value type "character".

valuemap.create

Description

object valuemap.create(object/array valuemaps)

This method allows to create new value maps.

Note:

This method is only available to Super admin user type. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parameters

(object/array) Value maps to create.

The method accepts value maps with the [standard value map properties](#).

Return values

(object) Returns an object containing the IDs of the created value maps the `valuemapids` property. The order of the returned IDs matches the order of the passed value maps.

Exemplos**Creating a value map**

Create one value map with two mappings.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "valuemap.create",
  "params": {
    "hostid": "50009",
    "name": "Service state",
    "mappings": [
      {
        "type": "1",
        "value": "1",
        "newvalue": "Up"
      },
      {
        "type": "5",
        "newvalue": "Down"
      }
    ]
  },
  "auth": "57562fd409b3b3b9a4d916d45207bbcb",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": {
    "valuemapids": [
      "1"
    ]
  },
  "id": 1
}
```

Fonte

CValueMap::create() in ui/include/classes/api/services/CValueMap.php.

valuemap.delete**Description**

object `valuemap.delete(array valuemapids)`

This method allows to delete value maps.

Note:

This method is only available to Super admin user type. Permissions to call the method can be revoked in user role settings. See [User roles](#) for more information.

Parâmetros

(array) IDs dos valores de mapa a serem deletados.

Return values

(object) Returns an object containing the IDs of the deleted value maps under the `valuemapids` property.

Exemplos

Excluindo múltiplos mapeamentos de valores

Deletar dois mapeamentos de valores.

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "valuemap.delete",
  "params": [
    "1",
    "2"
  ],
  "auth": "57562fd409b3b3b9a4d916d45207bbcb",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "valuemapids": [
      "1",
      "2"
    ]
  },
  "id": 1
}
```

Fonte

CValueMap::delete() in ui/include/classes/api/services/CValueMap.php.

valuemap.get**Descrição**

inteiro/array `valuemap.get`(parâmetros do objeto)

O método permite recuperar mapas de valor de acordo com os parâmetros fornecidos.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja [Funções de usuário](#) para maiores informações.

Parameters

(object) Parameters defining the desired output.

The method supports the following parameters.

Parameter	Type	Description
valuemapids	string/array	Return only value maps with the given IDs.
selectMappings	query	Return the value mappings for current value map in the mappings property.
sortfield	string/array	Supports count. Sort the result by the given properties.
countOutput	boolean	Possible values are: valuemapid , name . These parameters being common for all <code>get</code> methods are described in detail in the reference commentary .
editable	boolean	
excludeSearch	boolean	
filter	object	
limit	integer	
output	query	
preservekeys	boolean	
search	object	
searchByAny	boolean	
searchWildcardsEnabled	boolean	
sortorder	string/array	
startSearch	boolean	

Return values

(integer/array) Returns either:

- an array of objects;
- the count of retrieved objects, if the `countOutput` parameter has been used.

Exemplos

Retrieving value maps

Retrieve all configured value maps.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "valuemap.get",
  "params": {
    "output": "extend"
  },
  "auth": "57562fd409b3b3b9a4d916d45207bbcb",
  "id": 1
}
```

Response:


```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "valuemapid": "4",
      "name": "APC Battery Replacement Status"
    },
    {
      "valuemapid": "5",
      "name": "APC Battery Status"
    },
    {
      "valuemapid": "7",
      "name": "Dell Open Manage System Status"
    }
  ],
  "id": 1
}

```

Retrieve one value map with its mappings.

Request:

```

{
  "jsonrpc": "2.0",
  "method": "valuemap.get",
  "params": {
    "output": "extend",
    "selectMappings": "extend",
    "valuemapids": ["4"]
  },
  "auth": "57562fd409b3b3b9a4d916d45207bbcb",
  "id": 1
}

```

Response:

```

{
  "jsonrpc": "2.0",
  "result": [
    {
      "valuemapid": "4",
      "name": "APC Battery Replacement Status",
      "mappings": [
        {
          "type": "0",
          "value": "1",
          "newvalue": "unknown"
        },
        {
          "type": "0",
          "value": "2",
          "newvalue": "notInstalled"
        },
        {
          "type": "0",
          "value": "3",
          "newvalue": "ok"
        },
        {
          "type": "0",
          "value": "4",
          "newvalue": "failed"
        }
      ]
    }
  ]
}

```

```

        "type": "0",
        "value": "5",
        "newvalue": "highTemperature"
    },
    {
        "type": "0",
        "value": "6",
        "newvalue": "replaceImmediately"
    },
    {
        "type": "0",
        "value": "7",
        "newvalue": "lowCapacity"
    }
]
    }
],
    "id": 1
}

```

Fonte

CValueMap::get() in ui/include/classes/api/services/CValueMap.php.

valuemap.update

Descrição

object valuemap.update(object/array valuemaps)

Este método permite a atualização de mapas de valores existentes.

Note:

Este método está disponível apenas para o tipo de usuário Super admin. Permissões para chamar o método podem ser revogadas nas configurações de função do usuário. Para mais informações, consulte [User roles](#).

Parâmetros

Propriedades do mapa de valores a serem atualizadas (object/array) [Value map properties](#).

A propriedade valuemapid deve ser definida para cada mapa de valores, todas as outras propriedades são opcionais. Somente as propriedades passadas serão atualizadas, todas as outras permanecerão inalteradas.

Valores de retorno

(object) Retorna um objeto contendo os IDs dos mapas de valores atualizados na propriedade valuemapids.

Exemplos

Alterando o nome do mapa de valores

Altere o nome do mapa de valor para "Device status".

Requisição:

```

{
    "jsonrpc": "2.0",
    "method": "valuemap.update",
    "params": {
        "valuemapid": "2",
        "name": "Device status"
    },
    "auth": "57562fd409b3b3b9a4d916d45207bbcb",
    "id": 1
}

```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "valuemapids": [
      "2"
    ]
  },
  "id": 1
}
```

Alterando mapeamentos para um mapa de valores

Requisição:

```
{
  "jsonrpc": "2.0",
  "method": "valuemap.update",
  "params": {
    "valuemapid": "2",
    "mappings": [
      {
        "type": "0",
        "value": "0",
        "newvalue": "Online"
      },
      {
        "type": "0",
        "value": "1",
        "newvalue": "Offline"
      }
    ]
  },
  "auth": "57562fd409b3b3b9a4d916d45207bbcb",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "result": {
    "valuemapids": [
      "2"
    ]
  },
  "id": 1
}
```

Fonte

CValueMap::update() in ui/include/classes/api/services/CValueMap.php.

Verificação de descoberta

Esta classe foi projetada para trabalhar com verificações de descoberta.

Referências de objetos:

- [Descoberta check](#)

Métodos disponíveis:

- [dcheck.get](#) - recupera descoberta Verificações

> Objeto de verificação de descoberta

Os objetos a seguir estão diretamente relacionados à API dcheck.

Verificação de descoberta

O objeto de verificação de descoberta define uma verificação específica realizada por um regra de descoberta de rede. Tem as seguintes propriedades.

[Propriedade|**Tipo**|Descrição] |-----|-----| |dcheckid|string|(readonly) ID da verificação de descoberta. | |druleid|string|(readonly) ID da regra de descoberta à qual a verificação pertence. | |key_|string|O valor desta propriedade difere dependendo do tipo de verificação:
- chave para consultar as verificações do agente Zabbix, obrigatória;
- SNMP OID para verificações SNMPv1, SNMPv2 e SNMPv3, obrigatória. | |ports|string|Um ou vários intervalos de portas a serem verificados separados por vírgulas. Usado para todas as verificações, exceto para ICMP.

Padrão: 0. | |snmp_community|string|Comunidade SNMP.

Obrigatório para verificações de agente SNMPv1 e SNMPv2. | |snmpv3_authpassphrase|string|Senha de autenticação usada para verificações do agente SNMPv3 com nível de segurança definido como authNoPriv ou authPriv. | |snmpv3_authprotocol|integer|Protocolo de autenticação usado para verificações de agente SNMPv3 com nível de segurança definido como authNoPriv ou authPriv.

Valores possíveis:
0 - (padrão) MD5;
>1 - SHA1;
2 - SHA224;
3 - SHA256;
4 - SHA384;
5 - SHA512. | |snmpv3_contextname|string|nome do contexto SNMPv3. Usado apenas por verificações de SNMPv3. | |snmpv3_privpassphrase|string|Senha de privacidade usada para verificações do agente SNMPv3 com nível de segurança definido como authPriv. | |snmpv3_privprotocol|integer|Protocolo de privacidade usado para verificações de agente SNMPv3 com nível de segurança definido como authPriv.

Valores possíveis:
0 - (padrão) DES;
1 - AES128 ;
2 - AES192;
3 - AES256;
4 - AES192C;
5 - AES256C. | |snmpv3_securitylevel|string|Nível de segurança usado para verificações do agente SNMPv3.

Valores possíveis:
0 - noAuthNoPriv;
1 - authNoPriv;
2 - authPriv. | |snmpv3_securityname|string|Nome de segurança usado para verificações do agente SNMPv3. | **tipo**
(obrigatório)|inteiro|Tipo de verificação.

Valores possíveis:
0 - SSH;
1 - LDAP;
2 - SMTP;
3 - FTP;
4 - HTTP;
5 - POP;
6 - NNTP;
7 - IMAP;
8 - TCP;
9 - Agente Zabbix;
10 - Agente SNMPv1;
11 - Agente SNMPv2;
12 - Ping ICMP;
13 - Agente SNMPv3;
14 - HTTPS;
15 - Telnet. | |uniq|inteiro|Se usar esta verificação como critério de exclusividade do dispositivo. Apenas uma única verificação exclusiva pode ser configurada para uma regra de descoberta. Usado para verificações de agente Zabbix, SNMPv1, SNMPv2 e SNMPv3.

Valores possíveis:
0 - (padrão) não use esta verificação como critério de exclusividade;
1 - use esta verificação como critério de exclusividade. | |host_source|inteiro|Fonte para o nome do host.

Valores possíveis:
1 - (padrão) DNS;
2 - IP;
3 - valor de descoberta desta verificação. | |name_source|inteiro|Fonte para o nome visível.

Valores possíveis:
0 - (padrão) não especificado;
1 - DNS;
2 - IP;
>3 - valor de descoberta desta verificação. |

```
#####dcheck.get {#manual-api-reference-dcheck-get}
```

Descrição

```
integer/array dcheck.get(object parameters)
```

O método permite recuperar verificações de descoberta de acordo com o dado parâmetros.

Note:

Este método está disponível para usuários de qualquer tipo. Permissões para chamar o método pode ser revogado nas configurações de função do usuário. Veja **Usuário funções** Para maiores informações.

Parâmetros

(object) Parâmetros que definem a saída desejada.

O método oferece suporte aos seguintes parâmetros.

[Parâmetro|**Tipo**|Descrição] |-----|-----| |dcheckids|string/array|Retorna apenas verificações de descoberta com os IDs fornecidos. | |druleids|string/array|Retorna apenas verificações de descoberta que pertencem às regras de descoberta fornecidas. | |dserviceids|string/array|Retorna apenas verificações de descoberta que detectaram os serviços descobertos fornecidos. | |sortfield|string/array|Ordenar o resultado pelas propriedades fornecidas.

Os valores possíveis são: dcheckid e druleid. | |countOutput|boolean|Estes parâmetros sendo comuns para todos os métodos get são descritos em detalhes no **comentário de referência**. | |editável|boolean|^ | |excludePesquisa|boolean|^ | |filtro|objeto|^ | |limite|inteiro|^ | |saída|consulta|^ | |preservekeys|boolean|^ | |pesquisar|objeto|^ | |searchByAny|boolean|^ | |searchWildcardsEnabled|boolean|^ | |ordem de classificação|cadeia/matriz|^ | |startSearch|boolean|^ |

Valores de retorno

(integer/array) Retorna:

- uma matriz de objetos;

- a contagem de objetos recuperados, se o parâmetro `countOutput` tiver sido usado.

Exemplos

Recuperar verificações de descoberta para uma regra de descoberta

Recupere todas as verificações de descoberta usadas pela regra de descoberta "6".

Solicitação:

```
{
  "jsonrpc": "2.0",
  "método": "dcheck.get",
  "parâmetros": {
    "saída": "estender",
    "dcheckids": "6"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Resposta:

```
{
  "jsonrpc": "2.0",
  "resultado": [
    {
      "dcheckid": "6",
      "druleid": "4",
      "tipo": "3",
      "chave_": "",
      "snmp_community": "",
      "portas": "21",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "único": "0",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "host_source": "1",
      "name_source": "0"
    }
  ],
  "id": 1
}
```

Fonte

CDCheck::get() em `ui/include/classes/api/services/CDCheck.php`.

Apêndice 1. Comentário de referência

Notação Tipos de dados

A API do Zabbix suporta os seguintes tipos de dados como entrada:

Tipo	Descrição
boolean	Um valor booleano, aceita tanto <code>true</code> ou <code>false</code> .
flag	O valor é considerado <code>true</code> se for passado e não igual a <code>null</code> e <code>false</code> caso contrário.
integer	Um número inteiro.
float	Um número de ponto flutuante.
string	Uma string de texto.
text	Uma string de texto mais longa.

Tipo	Descrição
timestamp	Um timestamp Unix.
array	Uma sequência ordenada de valores, ou seja, uma matriz simples.
object	Uma array associativa.
query	Um valor que define quais dados devem ser retornados.
	Pode ser definido como uma matriz de nomes de propriedades para retornar apenas propriedades específicas ou como um dos valores predefinidos: extend - retorna todas as propriedades do objeto; count - retorna o número de registros recuperados, suportados apenas por algumas subseleções.

A API do Zabbix sempre retorna valores apenas como strings ou arrays.

Rótulos de propriedade

Algumas das propriedades dos objetos são marcadas com rótulos curtos para descrever

Seu comportamento. Os seguintes rótulos são usados:

- readonly - o valor do imóvel é definido automaticamente e não pode ser definido ou alterado pelo cliente;
- constant - o valor da propriedade pode ser definido ao criar um objeto, mas não pode ser alterado depois.

Valor de ID reservado "0" O valor de ID reservado "0" pode ser usado para filtrar elementos e remover objetos referenciados. Por exemplo, para remover um proxy referenciado de um host, proxy_hostid deve ser definido como 0 ("proxy_hostid": "0") ou para filtrar hosts monitorados pela opção do servidor proxyids deve ser definido como 0 ("proxyids": "0").

Parâmetros comuns do método "get" Os seguintes parâmetros são suportados por todos os métodos get:

Parâmetro	Tipo	Descrição
countOutput	boolean	Retorna o número de registros no resultado em vez dos dados reais.
editable	boolean	Se definido como true, retornará apenas objetos para os quais o usuário tem permissão de gravação.
excludeSearch	boolean	Default: false. Retorna resultados que não correspondem aos critérios fornecidos no parâmetro search.
filter	object	Retorne apenas os resultados que correspondem exatamente ao filtro fornecido. Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são um único valor ou uma matriz de valores para correspondência.
limit	integer	Não funciona para campos text. Limite o número de registros retornados.
output	query	Propriedades do objeto a serem retornadas.
preservekeys	boolean	Default: extend. Use IDs como chaves na matriz resultante.
search	object	Retornar resultados que correspondam à pesquisa de wildcard fornecida (case-insensitive). Aceita uma matriz, em que as chaves são nomes de propriedades e os valores são strings a serem pesquisadas. Se nenhuma opção adicional for fornecida, isso executará uma pesquisa LIKE "%...%".
searchByAny	boolean	Funciona apenas para campos string e text. Se definido como true retorna resultados que correspondem a qualquer um dos critérios fornecidos no parâmetro filter ou search em vez de todos eles.
searchWildcardsEnabled	boolean	Default: false. Se definido como true, habilita o uso de "*" como caractere curinga no parâmetro search.
		Default: false.

Parâmetro	Tipo	Descrição
sortfield	string/array	Classifique o resultado pelas propriedades fornecidas. Consulte uma descrição específica do método get da API para obter uma lista de propriedades que podem ser usadas para classificação. As macros não são expandidas antes da classificação.
sortorder	string/array	Se nenhum valor for especificado, os dados serão retornados sem classificação. Ordem de classificação. Se uma matriz for passada, cada valor será correspondido à propriedade correspondente fornecida no parâmetro sortfield. Os valores possíveis são: ASC - (default) ascendente; DESC - descendente.
startSearch	boolean	O parâmetro search irá comparar o início dos campos, ou seja, realizará uma pesquisa LIKE "...%". Ignorado se searchWildcardsEnabled estiver definido como true.

Exemplos Verificação de permissão do usuário

O usuário tem permissão para gravar em hosts cujos nomes começam com "MySQL" ou "Linux"?

Request:

```
{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "countOutput": true,
    "search": {
      "host": ["MySQL", "Linux"]
    },
    "editable": true,
    "startSearch": true,
    "searchByAny": true
  },
  "auth": "766b71ee543230a1182ca5c44d353e36",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": "0",
  "id": 1
}
```

Resultado zero significa que não há hosts com permissões de leitura/gravação.

Contagem de incompatibilidade

Conte o número de hosts cujos nomes não contêm a substring "ubuntu"

Request:

```
{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "countOutput": true,
    "search": {
      "host": "ubuntu"
    },
    "excludeSearch": true
  },
  "auth": "766b71ee543230a1182ca5c44d353e36",
  "id": 1
}
```

```
    "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": "44",
  "id": 1
}
```

Procurando por hosts usando wildcards

Encontre hosts cujo nome contenha a palavra "servidor" e tenham portas de interface "10050" ou "10071". Classifique o resultado por nome de host em ordem decrescente e limite-o a 5 hosts.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "host.get",
  "params": {
    "output": ["hostid", "host"],
    "selectInterfaces": ["port"],
    "filter": {
      "port": ["10050", "10071"]
    },
    "search": {
      "host": "*server*"
    },
    "searchWildcardsEnabled": true,
    "searchByAny": true,
    "sortfield": "host",
    "sortorder": "DESC",
    "limit": 5
  },
  "auth": "766b71ee543230a1182ca5c44d353e36",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "hostid": "50003",
      "host": "WebServer-Tomcat02",
      "interfaces": [
        {
          "port": "10071"
        }
      ]
    },
    {
      "hostid": "50005",
      "host": "WebServer-Tomcat01",
      "interfaces": [
        {
          "port": "10071"
        }
      ]
    },
    {
      "hostid": "50004",
      "host": "WebServer-Nginx",

```



```

        "interfaces": [
            {
                "port": "10071"
            }
        ],
        {
            "hostid": "99032",
            "host": "MySQL server 01",
            "interfaces": [
                {
                    "port": "10050"
                }
            ],
        },
        {
            "hostid": "99061",
            "host": "Linux server 01",
            "interfaces": [
                {
                    "port": "10050"
                }
            ],
        }
    ],
    "id": 1
}

```

Procurando hosts usando curingas com "preservekeys"

Se você adicionar o parâmetro "preservekeys" à requisição anterior, o resultado é retornado como um array associativo, onde as chaves são o id dos objetos.

Request:

```

{
    "jsonrpc": "2.0",
    "method": "host.get",
    "params": {
        "output": ["hostid", "host"],
        "selectInterfaces": ["port"],
        "filter": {
            "port": ["10050", "10071"]
        },
        "search": {
            "host": "*server*"
        },
        "searchWildcardsEnabled": true,
        "searchByAny": true,
        "sortfield": "host",
        "sortorder": "DESC",
        "limit": 5,
        "preservekeys": true
    },
    "auth": "766b71ee543230a1182ca5c44d353e36",
    "id": 1
}

```

Response:

```

{
    "jsonrpc": "2.0",
    "result": {
        "50003": {
            "hostid": "50003",

```

```

        "host": "WebServer-Tomcat02",
        "interfaces": [
            {
                "port": "10071"
            }
        ]
    },
    "50005": {
        "hostid": "50005",
        "host": "WebServer-Tomcat01",
        "interfaces": [
            {
                "port": "10071"
            }
        ]
    },
    "50004": {
        "hostid": "50004",
        "host": "WebServer-Nginx",
        "interfaces": [
            {
                "port": "10071"
            }
        ]
    },
    "99032": {
        "hostid": "99032",
        "host": "MySQL server 01",
        "interfaces": [
            {
                "port": "10050"
            }
        ]
    },
    "99061": {
        "hostid": "99061",
        "host": "Linux server 01",
        "interfaces": [
            {
                "port": "10050"
            }
        ]
    }
},
{id": 1
}

```

Apêndice 2. Mudanças de 5.4 para 6.0

Alterações incompatíveis com versões anteriores Ação¶

¶ Mudanças:¶ [ZBXNEXT-6755](#) action.create, action.update: Parâmetro renomeado de acknowledge_operations para update_operations.

[ZBXNEXT-6755](#) action.get: Parâmetro renomeado de selectAcknowledgeOperations para selectUpdateOperations.

[ZBXNEXT-6920](#) action.create, action.update: Adicionado validação restritivas para os parâmetros dos métodos.

Log de Auditoria¶

¶ Mudanças:¶ [ZBXNEXT-6715](#) Objeto de auditoria de log: descontinuado o suporte a propriedade 'nota'.

[ZBXNEXT-6715](#) Objeto de auditoria de log: descontinuado o suporte ao valor 2 do resourcetype (Configuração do Zabbix) e ao

7 (Elemento gráfico).

[ZBXNEXT-6715](#) Objeto de auditoria de log: descontinuado o suporte ao valor 5 do action (Habilitar) e 6 (Desabilitar).

[ZBXNEXT-6715](#) auditlog.get: descontinuado o suporte ao parâmetro selectDetails.

[ZBXNEXT-6718](#) Objeto de auditoria de log: descontinuado o suporte ao valor 3 da action (Login).

grupo de host

Mudanças:

[ZBXNEXT-6868](#) hostgroup.massupdate: hosts etemplates os campos agora são obrigatórios.

[ZBXNEXT-6868](#) hostgroup.massadd, hostgroup.massupdate, hostgroup.massremove: adicionada validação estrita dos parâmetros do método.

host prototype

Alterações:

[ZBXNEXT-6959](#) hostprototype.get: suporte removido para as propriedades group_prototypeid, hostid, templateid dos objetos de API de vínculo de grupo e protótipo de grupos.

[ZBXNEXT-6959](#) hostprototype.update: removida a capacidade de modificar os campos somente leitura host, name, custom_interfaces, interfaces, groupLinks, groupPrototypes, templates, tags, macros, inventory_mode de protótipos de hosts herdados.

[ZBXNEXT-6959](#) hostprototype.create, hostprototype.update, hostprototype.delete: adição de validação estrita dos parâmetros do método.

mapa de ícones

Mudanças:

[ZBXNEXT-6914](#) iconmap.create, iconmap.update: suporte descartado da propriedade do objeto de mapeamento de ícone sortorder.

manutenção

Mudanças:

[ZBXNEXT-6890](#) maintenance.create, maintenance.update, maintenance.delete: adicionada validação estrita dos parâmetros dos métodos.

[ZBXNEXT-6890](#) maintenance.get, maintenance.update: suporte descartado para parâmetro timeperiodid do objeto de período de tempo.

tipo de mídia

Mudanças:

[ZBXNEXT-6885](#) mediatype.create, mediatype.update: adicionada validação estrita dos parâmetros dos métodos.

Função

Mudanças:

[ZBXNEXT-6787](#) dropped support of value manage_services for name property in action object.

[ZBXNEXT-3022](#) descartado o suporte de valor configuration.services por nome propriedade no objeto UI.

serviço

Alterações:

[ZBXNEXT-6999](#) adicionou as propriedades uuid, description e created_at .

[ZBXNEXT-6999](#) removeu o suporte para as propriedades showsla, goodsla e times .

[ZBXNEXT-6800](#) alterou o valor do status de "0" de "OK" para "Não classificado".

[ZBXNEXT-3022](#) removeu o suporte para service.adddependencies, service.addtimes, service.deletedependencies, service.deletetimes.

[ZBXNEXT-6674](#) removeu o suporte para a propriedade triggerid.

[ZBXNEXT-6999](#) service.get: removeu o suporte para os parâmetros showsla, selectAlarms, selectTimes .

[ZBXNEXT-6999](#) service.getsla: removeu o suporte para o método.

[ZBXNEXT-6999](#) métodos sla.get, sla.create, sla.update, sla.delete, sla.getsli adicionados.

[ZBXNEXT-6999](#) service.get: adicionou suporte para ordenação por serviceid, status e created_at.

[ZBXNEXT-6999](#) `service.get`: adicionou suporte para o parâmetro `slaids` ; adicionou suporte para filtragem por `uuid`.
[ZBXNEXT-6999](#) `service.create`, `service.update`: removeu suporte para parâmetros `showsla`, `goodsla` e `times` .
[ZBXNEXT-2406](#) `service.getsla`: removeu as propriedades `status` e `problemsa` resposta da solicitação com o parâmetro `intervals` .
[ZBXNEXT-3022](#) `service.create`, `service.update`: removeu o suporte para os parâmetros `dependencies` e `parentid`.
[ZBXNEXT-3022](#) `service.get`: removeu o suporte para os parâmetros `selectParent`, `selectDependencies` e `selectParentDependencies`.
[ZBXNEXT-6674](#) `service.get`: removeu o suporte para parâmetro `selectTrigger`.

template

Mudanças:

[ZBXNEXT-6867](#) `template.create`, `template.update`, `template.delete`, `template.massadd`, `template.massupdate`, `template.massremove`: adicionada validação estrita dos parâmetros dos métodos.
[ZBXNEXT-6867](#) `template.create`, `template.update`, `template.massadd`, `template.massupdate`: descartado o suporte para o parâmetro `hosts`.
[ZBXNEXT-6867](#) `template.massremove`: descartado o suporte para o parâmetro `hostids`.

trigger

Mudanças:

[ZBXNEXT-6867](#) `trigger.adddependencies`, `trigger.deletedependencies`: eliminada a capacidade de editar as dependências de gatilhos herdados.

Outras alterações e correção de bugs ação

Mudanças:

[ZBXNEXT-6755](#) adicionados novos valores `conditiontype` (27 - Service, 28 - Service name).
[ZBXNEXT-6250](#) `action.get`, `action.create`, `action.update`: adicionou nova propriedade `notify_if_canceled`.

log de auditoria

Mudanças:

[ZBXNEXT-6999](#) adicionado novo `resourcetype` (48 - SLA).
[ZBXNEXT-6923](#) adicionado novo `resourcetype` (47 - High availability node).
[ZBXNEXT-6718](#) adicionado suporte aos valores `action` (8 - Login, 9 - Failed login, 10 - History clear).
[ZBXNEXT-6715](#) adicionado suporte as propriedades: `username`, `recordsetid`, `details`.

autenticação

Mudanças:

[ZBXNEXT-4029](#) adicionados novos campos de política de senha `passwd_min_length` e `passwd_check_rules`.

dashboard

Mudanças:

[ZBXNEXT-6999](#) adicionado novo tipo de widget `slareport` e tipos de campo de widget (9 - Service, 10 - SLA).
[ZBXNEXT-6966](#) adicionado novo tipo de widget `item`.

histórico

Mudanças:

[ZBXNEXT-6714](#) adicionado novo método de API `history.clear`.

housekeeping

Mudanças:

[ZBXNEXT-6755](#) adicionado suporte da propriedade `hk_events_service`.

item prototype

Changes:

[ZBXNEXT-7049](#) `itemprototype.get`, `itemprototype.create`, `itemprototype.update`: adicionado o terceiro parâmetro ao pré-processamento no padrão Prometheus. O segundo parâmetro agora determinará um método de agregação: `value`,

label, function. O terceiro parâmetro agora conterá a saída do Prometheus para o método de agregação label ou a função de agregação para o método de agregação function.

manutenção

Mudanças:

[ZBXNEXT-6890](#) maintenance.create, maintenance.update: o parâmetro groupids está obsoleto. Em vez disso, use groups.

[ZBXNEXT-6890](#) maintenance.create, maintenance.update: o parâmetro hostids está obsoleto. Em vez disso, use hosts.

[ZBXNEXT-6890](#) maintenance.create, maintenance.update: mudou o tipo de parâmetros groups, hosts, timeperiods e tags de array para object/array.

mediatype

Mudanças:

[ZBXNEXT-6755](#) objeto de modelo de mensagem: adicionado suporte do valor conditiontype: 4 - (evento criado na atualização do status do serviço).

proxy

Mudanças:

[ZBXNEXT-6889](#) proxy.create, proxy.update: removido suporte as propriedades interface.interfaceid e interface.hostid.

regex

Mudanças:

[ZBXNEXT-6717](#) adicionado API regex.get, regex.create, regex.update e regex.delete.

função

Mudanças:

[ZBXNEXT-6787](#) adicionado suporte as novas propriedades services.read.mode, services.read.list, services.read.tag, services.write.mode, services.write.list e services.write.tag.

[ZBXNEXT-3022](#) suporte adicionado ao valor do nome manage_services.

serviço

Mudanças:

[ZBXNEXT-6787](#) adicionada nova propriedade readonly.

[ZBXNEXT-6800](#) adicionada novas propriedades weight, propagation_rule e propagation_value.

[ZBXNEXT-6800](#) suporte adicionado ao valor de status (-1 - OK).

[ZBXNEXT-2406](#) service.get: adicionado suporte aos parâmetros deep_parentids e selectProblemEvents.

[ZBXNEXT-6800](#) service.create, service.update: adicionado suporte ao parâmetro status_rules.

[ZBXNEXT-6800](#) service.get: adicionado suporte ao parâmetro selectStatusRules.

[ZBXNEXT-6800](#) service.get: adicionado suporte de count para o parâmetro selectAlarms.

[ZBXNEXT-3022](#) service.create, service.update: adicionado suporte aos parâmetros children, parents e tags.

[ZBXNEXT-3022](#) service.get: adicionado suporte aos parâmetros evaltype, tags, selectChildren, selectParents, selectTags.

[ZBXNEXT-3022](#) service.get: adicionado suporte aos parâmetros problem_tags, without_problem_tags and selectProblemTags.

[ZBXNEXT-6674](#) service.create, service.update: adicionado suporte ao parâmetro problem_tags.

configurações

Mudanças:

[ZBXNEXT-6945](#) settings.get, settings.update: adicionado suporte aos parâmetros geomaps_tile_provider, geomaps_tile_url, geomaps_max_zoom e geomaps_attribution.

[ZBXNEXT-6715](#) settings.get, settings.update: adicionado suporte ao parâmetro auditlog_enabled.

sla

Mudanças:

[ZBXNEXT-6999](#) adicionada uma nova API sla com métodos: sla.create, sla.delete, sla.get, sla.getsli, sla.update.

templatedashboard

Mudanças:

[ZBXNEXT-6966](#) adicionado suporte de novo tipo de widget item.

usuário

Mudanças:

[ZBXNEXT-6718](#) adicionado novo método `user.unblock`.

[ZBXNEXT-4029](#) `user.create`, `user.update`: validação de força de senha implementada de acordo com a política de senha.

grupo de usuários

Mudanças:

[ZBXNEXT-6866](#) `usergroup.create`, `usergroup.update`: `userids` o parâmetro agora está obsoleto. Use `users` em vez disso.

item

Mudanças:

[ZBXNEXT-7049](#) `item.get`, `item.create`, `item.update`: adicionou o terceiro parâmetro à etapa de pré-processamento do padrão Prometheus. O segundo parâmetro agora determinará um método de agregação: `value`, `label`, `function`. O terceiro parâmetro agora conterá a saída do Prometheus para o método de agregação `label` ou uma função de agregação para o método de agregação `function`.

Mudanças na API do Zabbix na versão 6.0

6.0.19 ação

Mudanças:

[ZBX-21804](#) `action.create`, `action.get`, `action.update`, `action.delete`: Removido o requisito de permissões de gravação para hosts, grupos de hosts, templates, triggers, proxy.

6.0.15 script

Mudanças:

[ZBX-19466](#) Validação de alteração de objeto de script para ser único pela combinação de 2 propriedades: `name` e `menu_path`.

6.0.14 usuário

Mudanças:

[ZBXNEXT-8012](#) `user.checkAuthentication`: novo parâmetro adicionado: `token`.

6.0.13 configuração

Mudanças:

[ZBXNEXT-7951](#) `configuration.import`, `configuration.importcompare`: opções `deleteMissing: true` para `templateLinkage` will now unlink missing templates (instead of unlink and clear).

discoveryrule

Mudanças:

[ZBXNEXT-7951](#) `discoveryrule.update`: o parâmetro `uuid` pode agora ser atualizado.

graph

Mudanças:

[ZBXNEXT-7951](#) `graph.update`: o parâmetro `uuid` pode agora ser atualizado.

graphprototype

Mudanças:

[ZBXNEXT-7951](#) graphprototype.update: o parâmetro uuid pode agora ser atualizado.

hostgroup

Mudanças:

[ZBXNEXT-7951](#) hostgroup.update: o parâmetro uuid pode agora ser atualizado.

hostprototype

Mudanças:

[ZBXNEXT-7951](#) hostprototype.update: o parâmetro uuid pode agora ser atualizado.

httptest

Mudanças:

[ZBXNEXT-7951](#) httptest.update: o parâmetro uuid pode agora ser atualizado.

item

Mudanças:

[ZBXNEXT-7951](#) item.update: o parâmetro uuid pode agora ser atualizado.

itemprototype

Mudanças:

[ZBXNEXT-7951](#) itemprototype.update: o parâmetro uuid pode agora ser atualizado.

template

Mudanças:

[ZBXNEXT-7951](#) template.update: o parâmetro uuid pode agora ser atualizado.

templatedashboard

Mudanças:

[ZBXNEXT-7951](#) templatedashboard.update: o parâmetro uuid pode agora ser atualizado.

trigger

Mudanças:

[ZBXNEXT-7951](#) trigger.update: o parâmetro uuid pode agora ser atualizado.

triggerprototype

Mudanças:

[ZBXNEXT-7951](#) triggerprototype.update: o parâmetro uuid pode agora ser atualizado.

valuemap

Mudanças:

[ZBXNEXT-7951](#) valuemap.update: o parâmetro uuid pode agora ser atualizado.

6.0.9 user

Mudanças:

[ZBXNEXT-7971](#) user.create, user.update: comprimento máximo do campo "url" aumentado para 2.048 caracteres.

6.0.7 gráfico

Mudanças:

[ZBX-7706](#) `graph.get`: A disponibilidade do gráfico não depende de permissões para itens especificados nos campos "ymin_itemid" e "ymax_itemid".

O gráfico com o eixo MIN ou MAX Y vinculado a itens inacessíveis ainda estará acessível, mas o eixo MIN/MAX Y funciona da mesma maneira como se o método de cálculo especificado fosse "Calculated".

gráficoprotótipo

Mudanças:

[ZBX-7706](#) `graphprototype.get`: A disponibilidade do protótipo de gráfico não depende de permissões para itens especificados nos campos de protótipo de gráfico "ymin_itemid" e "ymax_itemid".

6.0.3 regradedescoberta

Correções de bugs:

[ZBX-19118](#) `discoveryrule.create`, `discoveryrule.update`: A propriedade `interfaceid` não é mais necessária para criar/atualizar uma regra LLD do tipo de agente HTTP.

item

Correções de bugs:

[ZBX-19118](#) `item.create`, `item.update`: A propriedade `interfaceid` não é mais necessária para criar/atualizar um item do tipo de agente HTTP.

itemprotótipo

Correções de bugs:

[ZBX-19118](#) `itemprototype.create`, `itemprototype.update`: A propriedade `interfaceid` não é mais necessária para criar/atualizar um protótipo de item do tipo de agente HTTP.

20. Módulos

Visão geral É possível melhorar a funcionalidade do frontend do Zabbix adicionando 3º módulos do partido ou desenvolvendo seus próprios módulos sem a necessidade de alterar o código fonte do Zabbix.

Observe que o código do módulo será executado com os mesmos privilégios do Zabbix Código fonte. Isso significa:

- Módulos de terceiros podem ser prejudiciais. Você deve confiar nos módulos que você está instalação;
- Erros em um código de módulo de terceiros podem travar o frontend. Se este acontece, basta remover o código do módulo do frontend. Assim que você recarregar o frontend do Zabbix, você verá uma nota dizendo que alguns módulos estão ausentes. Vá para **Módulo administração** (em Administração → Geral → Módulos) e clique em Digitalizar directory novamente para remover módulos inexistentes do banco de dados.

Instalação Por favor, leia sempre o manual de instalação de um módulo específico. Isto é recomendado instalar novos módulos um por um para detectar falhas facilmente.

Pouco antes de instalar um módulo:

- Certifique-se de ter baixado o módulo de uma fonte confiável. A instalação de código nocivo pode levar a consequências, como dados perda
- Diferentes versões do mesmo módulo (mesmo ID) podem ser instaladas em paralelo, mas apenas uma única versão pode ser habilitada por vez

Etapas para instalar um módulo:

- Descompacte seu módulo dentro de sua própria pasta na pasta `modules` da interface do Zabbix
- Certifique-se de que a pasta do módulo contenha pelo menos o `manifest.json` Arquivo
- Navegue até **Módulo administração** e clique no botão Digitalizar diretório
- Novo módulo aparecerá na lista junto com sua versão, autor, descrição e estado
- Habilite o módulo clicando em seu status

Solução de problemas:

Problema	Solução
O módulo não apareceu na lista	Certifique-se de que o arquivo manifest.json existe na pasta <code>modules/your-module/</code> do frontend do Zabbix. Se isso acontecer, significa que o módulo não é compatível com a versão atual do Zabbix. Se o arquivo manifest.json não existir, você provavelmente descompactou no diretório errado.
Frontend travado	O código do módulo não é compatível com a versão atual do Zabbix ou configuração do servidor. Por favor, exclua os arquivos do módulo e recarregue o frontend. Você verá um aviso de que alguns módulos estão ausentes. Vá para Module administration e clique em Scan directory novamente para remover módulos inexistentes do banco de dados.
Aparece uma mensagem de erro sobre namespace, ID ou ações idênticas	Novo módulo tentou registrar um namespace, ID ou ações que já estão registradas por outros módulos habilitados. Desabilite o módulo conflitante (mencionado na mensagem de erro) antes de habilitar o novo.
Aparecem mensagens de erro técnico	Relatar erros ao desenvolvedor do módulo.

Desenvolvendo módulos Os módulos são escritos em linguagem PHP. Controlador de visualização de modelo (MVC) design de padrão de software é o preferido, pois também é usado no Zabbix frontend e facilitará o desenvolvimento. A tipagem estrita do PHP também é bem-vindo, mas não obrigatório.

Observe que com os módulos você pode adicionar facilmente novos itens de menu e respectivas visualizações e ações para o frontend do Zabbix. Atualmente não é possível registrar uma nova API ou criar novas tabelas de banco de dados através módulos.

Estrutura do módulo

Cada módulo é um diretório (colocado dentro do diretório `modules`) com subdiretórios contendo controllers, views e qualquer outro código:

```
example_module_directory/ (obrigatório)
  manifest.json (obrigatório) Metadados e definição de ação.
  Module.php Inicialização do módulo e manipulação de eventos.
  actions/Arquivos do controlador de ação.
    AlgoView.php
    AlgoCreate.php
    AlgoDelete.php
  data_export/
    ExportAsXml.php
    ExportAsExcel.php
  visualizações/Visualizar arquivos.
    exemplo.algo.view.php
    exemplo.algo.excluir.php
  js/ arquivos JavaScript usados em views.
    exemplo.algo.view.js.php
  parciais/ Visualizar arquivos parciais.
    exemplo.algo.reutilizável.php
  js/ arquivos JavaScript usados em parciais.
    exemplo.algo.reutilizável.js.php
```

Como você pode ver, o único arquivo obrigatório dentro do módulo personalizado diretório é `manifest.json`. O módulo não será registrado sem isso Arquivo. `Module.php` é responsável por registrar itens de menu e processamento de eventos como `'onBeforeAction'` e `'onTerminate'`. O Os diretórios `actions`, `views` e `partials` contêm PHP e JavaScript código necessário para ações do módulo.

Convenção de nomes

Antes de criar um módulo, é importante concordar com a nomenclatura convenção para diferentes itens de módulo, como diretórios e arquivos, que poderíamos manter as coisas bem organizadas. Você também pode encontrar exemplos acima, na seção **Estrutura do módulo**.

Item	Regras de nomenclatura	Exemplo
Diretório do módulo	minúsculas [a-z], sublinhado e dígitos decimais	exemplo_v2
Subdiretórios de ação	minúsculas [a-z] e caractere de sublinhado	dados_export
Arquivos de ação	CamelCase, terminando com tipo de ação	SomethingView.php
Visualização e arquivos parciais	minúsculas [az] Palavras separadas por ponto Prefixado por <code>module.</code> seguido pelo nome do módulo Terminando com tipo de ação e extensão de arquivo <code>.php</code>	module.example.algo.view.php
Arquivos Javascript	As mesmas regras se aplicam aos arquivos de visualização e parciais, exceto a extensão de arquivo <code>.js.php</code> .	module.example.something.view.js.php

Observe que o prefixo 'módulo' e a inclusão do nome são obrigatórios para visualização e nomes de arquivos parciais, a menos que você precise substituir as visualizações principais do Zabbix ou parciais. Essa regra, no entanto, não se aplica a nomes de arquivos de ação.

Preparação do manifesto

Espera-se que cada módulo tenha um arquivo `manifest.json` com o seguinte campos no formato JSON:

Parâmetro	Requerido	Tipo	Padrão	Descrição
<code>manifest_version</code>	Yes	Double	-	Versão de manifesto do módulo. A versão atualmente suportada é 1 .
<code>id</code>	Sim	String	-	ID do módulo. Apenas um módulo com determinado ID pode ser habilitado ao mesmo tempo.
<code>name</code>	Yes	String	-	Nome do módulo conforme exibido na seção Administração.
<code>versão</code>	Sim	String	-	Versão do módulo conforme exibido na seção Administração.
<code>namespace</code>	Yes	String	-	PHP namespace para <code>Module.php</code> e classes de ação.
<code>autor</code>	Não	String	""	Autor do módulo conforme exibido na seção Administração.
<code>url</code>	No	String	""	URL do módulo conforme exibido na seção Administração.
<code>description</code>	No	String	""	Descrição do módulo conforme exibido na seção Administração.
<code>actions</code>	No	Object	{}	Ações para registrar com este módulo. Consulte Ações.
<code>config</code>	No	Object	{}	Configuração do módulo.

Para referência, veja um exemplo de `manifest.json` no seção [Reference](#).

Ações

O módulo terá controle sobre as ações de frontend definidas dentro do `actions` no arquivo `manifest.json`. Desta forma, novas ações são definidas. Da mesma forma, você pode redefinir as ações existentes. Cada chave de `actions` devem representar o nome da ação e o valor correspondente deve conter as chaves `class` e opcionalmente `layout` e `view`.

Uma ação é definida por quatro contrapartes: nome, controlador, visão e disposição. A validação e a preparação dos dados geralmente são feitas no controlador, a formatação de saída é feita na visão ou parciais, e o layout é responsável por decorar a página com elementos como menu, cabeçalho, rodapé e outros.

As ações do módulo devem ser definidas no arquivo `manifest.json` como `actions` objeto:

Parâmetro	Requerido	Tipo	Padrão	Descrição
chave	Sim	String	-	Nome da ação, em letras minúsculas [a-z], separando palavras com ponto.
class	Yes	String	-	Nome da classe de ação, incluindo o caminho do subdiretório (se usado) dentro do diretório actions.
layout	Não	String	"layout.htmlpage"	Layout da ação.
view	No	String	null	Visualização de ação.

Existem vários layouts predefinidos, como `layout.json` ou `layout.xml`. Estes destinam-se a ações que produzem diferentes resultado do que um HTML. Você pode explorar layouts predefinidos no `app/views/` diretório ou até mesmo criar o seu próprio.

Às vezes é necessário apenas redefinir a visão parte de alguma ação deixando o controlador intacto. Nesse caso, basta colocar o necessário view e/ou arquivos parciais dentro do diretório `views` do módulo.

Para referência, consulte um exemplo de arquivo de controlador de ação no seção [Reference](#). Por favor, não hesite em explorar ações atuais do código fonte do Zabbix, localizadas no diretório `app/`.

Módulo.php

Este arquivo PHP opcional também é responsável pela inicialização do módulo como tratamento de eventos. Espera-se que a classe 'Módulo' seja definida neste arquivo, estendendo a classe base `\Core\CModule`. A classe Módulo deve ser definido no namespace especificado no arquivo `manifest.json`.

```
<?php

Módulos de namespace\Exemplo;
use Core\CModule como BaseModule;

class Module estende BaseModule {
    ...
}
```

Para referência, veja um exemplo de `Module.php` no seção [Reference](#).

Referência Esta seção contém versões básicas de diferentes elementos do módulo introduzido nas seções anteriores.

manifest.json

```
{
  "manifest_version": 1.0,
  "id": "example_module",
  "name": "Example module",
  "version": "1.0",
  "namespace": "Example",
  "author": "John Smith",
  "url": "http://module.example.com",
  "description": "Short description of the module.",
  "actions": {
    "example.something.view": {
      "class": "SomethingView",
      "view": "module.example.something.view"
    },
    "example.something.create": {
      "class": "SomethingCreate",
      "layout": null
    },
    "example.something.delete": {
      "class": "SomethingDelete",
      "layout": null
    },
    "example.something.export.xml": {
      "class": "data_export/ExportAsXml",

```

```

        "layout": null
    },
    "example.something.export.excel": {
        "class": "data_export/ExportAsExcel",
        "layout": null
    }
},
"config": {
    "username": "john_smith"
}
}

```

Módulo.php

```

<?php declare(strict_types = 1);

namespace Modules\Example;

use APP;
use CController as CAction;

/**
 * Please see Core\CModule class for additional reference.
 */
class Module extends \Core\CModule {

    /**
     * Initialize module.
     */
    public function init(): void {
        // Initialize main menu (CMenu class instance).
        APP::Component()->get('menu.main')
            ->findOrAdd(_('Reports'))
                ->getSubmenu()
                    ->add((new \CMenuItem(_('Example wide report'))))
                        ->setAction('example.report.wide.php')
                    )
                ->add((new \CMenuItem(_('Example narrow report'))))
                    ->setAction('example.report.narrow.php')
                );
    }

    /**
     * Event handler, triggered before executing the action.
     *
     * @param CAction $action Action instance responsible for current request.
     */
    public function onBeforeAction(CAction $action): void {
    }

    /**
     * Event handler, triggered on application exit.
     *
     * @param CAction $action Action instance responsible for current request.
     */
    public function onTerminate(CAction $action): void {
    }
}

```

Controlador de ação

```

<?php declare(strict_types = 1);

namespace Modules\Example\Actions;

```

```

use CControllerResponseData;
use CControllerResponseFatal;
use CController as CAction;

/**
 * Example module action.
 */
class SomethingView extends CAction {

    /**
     * Initialize action. Method called by Zabbix core.
     *
     * @return void
     */
    public function init(): void {
        /**
         * Disable SID (Session ID) validation. Session ID validation should only be used for actions which
         * modification, such as update or delete actions. In such case Session ID must be presented in the
         * the URL would expire as soon as the session expired.
         */
        $this->disableSIDvalidation();
    }

    /**
     * Check and sanitize user input parameters. Method called by Zabbix core. Execution stops if false is
     *
     * @return bool true on success, false on error.
     */
    protected function checkInput(): bool {
        $fields = [
            'name' => 'required|string',
            'email' => 'required|string',
            'phone' => 'string'
        ];

        // Only validated data will further be available using $this->hasInput() and $this->getInput().
        $ret = $this->validateInput($fields);

        if (!$ret) {
            $this->setResponse(new CControllerResponseFatal());
        }

        return $ret;
    }

    /**
     * Check if the user has permission to execute this action. Method called by Zabbix core.
     * Execution stops if false is returned.
     *
     * @return bool
     */
    protected function checkPermissions(): bool {
        $permit_user_types = [USER_TYPE_ZABBIX_ADMIN, USER_TYPE_SUPER_ADMIN];

        return in_array($this->getUserType(), $permit_user_types);
    }

    /**
     * Prepare the response object for the view. Method called by Zabbix core.
     *
     * @return void
     */

```

```

protected function doAction(): void {
    $contacts = $this->getInput('email');

    if ($this->hasInput('phone')) {
        $contacts .= ', ' . $this->getInput('phone');
    }

    $data = [
        'name' => $this->getInput('name'),
        'contacts' => $contacts
    ];

    $response = new CControllerResponseData($data);

    $this->setResponse($response);
}
}

```

Visualização de ação

```

<?php declare(strict_types = 1);

/**
 * @var CView $this
 */

$this->includeJsFile('example.something.view.js.php');

(new CWidget())
->setTitle(_('Something view'))
->addItem(new CDiv($data['name']))
->addItem(new CPartial('module.example.something.reusable', [
    'contacts' => $data['contacts']
]))
->show();

```

21. Apêndices

Por favor, use a barra lateral para acessar o conteúdo na seção Apêndices.

1 Perguntas frequentes / Solução de problemas

Perguntas frequentes ou FAQ.

1. P: Posso liberar/limpar a fila (conforme descrito em Administração → Fila)?
R: Não.
2. P: Como faço para migrar de um banco de dados para outro?
R: Apenas dados de despejo (para MySQL, use sinalizador -t ou --no-create-info), crie o novo banco de dados usando arquivos de esquema do Zabbix e importe os dados.
3. P: Gostaria de substituir todos os espaços por sublinhados no meu item keys porque funcionavam em versões mais antigas, mas o espaço não é válido símbolo para uma chave de item em 3.0 (ou qualquer outro motivo para modificar em massa teclas de itens). Como devo fazer e com o que devo tomar cuidado?
R: Você pode usar uma consulta de banco de dados para substituir todas as ocorrências de espaços nas chaves do item com sublinhados:
atualizar itens set key_=replace(key_',','_');
Os gatilhos poderão usar esses itens sem nenhuma modificações, mas você pode ter que alterar qualquer referência de item em estes locais:
* Notificações (ações)

* Elemento de mapa e rótulos de link

* Fórmulas de itens calculados

4. P: Meus gráficos têm pontos em vez de linhas ou áreas vazias. Por quê então?

R: Faltam dados. Isso pode acontecer por vários motivos - problemas de desempenho no banco de dados Zabbix, servidor Zabbix, rede, dispositivos monitorados...

5. P: Os daemons do Zabbix falham ao iniciar com uma mensagem Falha no ouvinte com erro: socket() para [:-10050] falhou com erro 22: Argumento inválido.

R: Este erro surge na tentativa de executar o agente Zabbix compilado em versão 2.6.27 ou superior em uma plataforma com kernel 2.6.26 e mais baixo. Observe que a vinculação estática não ajudará neste caso porque é a chamada de sistema socket() que não suporta SOCK_CLOEXEC sinalizar em kernels anteriores. [ZBX-3395](#)

6. P: Tento configurar um parâmetro de usuário flexível (um que aceite parâmetros) com um comando que usa um parâmetro posicional como \$1, mas não funciona (utiliza o parâmetro item). Como resolver esta?

R: Use um cifrão duplo como **\$\$1**

7. P: Todos os menus suspensos têm uma barra de rolagem e ficam feios no Opera 11. Por que assim?

R: É um bug conhecido no Opera 11.00 e 11.01; veja [problema Zabbix rastreador](#) para mais em formação.

8. P: Como posso alterar a cor de fundo do gráfico em um tema personalizado?

R: Veja a tabela graph_theme no banco de dados e [theming guia](#).

9. P: Com DebugLevel 4 estou vendo mensagens "Trapper got [] len 0" em log do servidor/proxy - o que é isso?

R: Provavelmente é frontend, conectando e verificando se servidor ainda está em execução.

10. P: Meu sistema tinha a hora definida no futuro e agora nenhum dado é chegando. Como isso poderia ser resolvido?

R: Limpar valores dos campos do banco de dados hosts.disable_until*, drules.nextcheck, httptest.nextcheck e reinicie o servidor/proxy.

11. P: Valores de itens de texto no frontend (ao usar a macro {ITEM.VALUE} e em outros casos) são cortados/aparados para 20 símbolos. Isso é normal?

R: Sim, há um limite codificado em include/items.inc.php Atualmente.

Se você não encontrou resposta para sua pergunta tente [Zabbix fórum](#)

2 Instalação e configuração

1 Criação do banco de dados

Visão geral

Um banco de dados Zabbix deve ser criado durante a instalação do Zabbix Server ou Proxy.

Esta seção fornece instruções para criação de um banco de dados Zabbix. Um conjunto separado de instruções está disponível para cada banco de dados suportado.

O padrão UTF-8 é o único suportado pelo Zabbix. É sabido trabalhar sem quaisquer falhas de segurança. Os usuário devem estar cientes de que há problemas de segurança conhecidos ao usar outros padrões de codificação.

Note:

Se você instalar a partir do repositório do Zabbix [Zabbix Git repository](#), você precisa executar o seguinte comando e então ir para os próximos passos:

```
<br><br> make dbschema
```

MySQL

Configurações com utf8 (aka utf8mb3) e utf8mb4 são suportadas (com collation utf8_bin and utf8mb4_bin respectivamente) para servidor/proxy do Zabbix para funcionarem com banco de dados MySQL. É recomendado utilizar utf8mb4 para novas instalações.

Para o Zabbix 6.0.11 e versões mais recentes, triggers determinísticas precisam ser criadas durante a importação do schema. No MySQL e MariaDB, é necessário que GLOBAL log_bin_trust_function_creators = 1 seja configurado caso o logging binário esteja habilitado e não haja privilégios para superuser, além de log_bin_trust_function_creators = 1 não estar definido no arquivo de configuração MySQL.

Se você estiver instalando a partir dos **pacotes** do Zabbix, vá para instruções [RHEL/CentOS](#).

Se você estiver instalando o Zabbix a partir de fontes:

- Crie e configure um banco de dados e um usuário.

```
mysql -uroot -p<password>
```

```
mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;
mysql> create user 'zabbix'@'localhost' identified by '<password>';
mysql> grant all privileges on zabbix.* to 'zabbix'@'localhost';
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
mysql> quit;
```

- Importe os dados para o banco de dados. Para um banco de dados do Zabbix proxy, apenas o `schema.sql` deve ser importado (não `images.sql` nem `data.sql`).

```
cd database/mysql
mysql -uzabbix -p<password> zabbix < schema.sql
#### stop here if you are creating database for Zabbix proxy
mysql -uzabbix -p<password> zabbix < images.sql
mysql -uzabbix -p<password> zabbix < data.sql
```

O `log_bin_trust_function_creators` pode ser desabilitado assim que o `schema` tiver sido importado com sucesso:

```
mysql -uroot -p<password>

mysql> SET GLOBAL log_bin_trust_function_creators = 0;
mysql> quit;
```

PostgreSQL

Você precisa ter um usuário do banco de dados com permissão de criação de objetos de banco.

Se você estiver instalando a partir de **pacotes** do Zabbix, siga as [instruções](#) para a sua plataforma.

Se você estiver instalando a partir o Zabbix a partir das fontes:

- Crie um usuário para o banco de dados

O comando shell a seguir criará o usuário `zabbix`. Especifique uma senha quando solicitado e repita a senha (observe que primeiro deve ser solicitado que você informe a senha de `sudo`):

```
sudo -u postgres createuser --pwprompt zabbix
```

- Crie um banco de dados

O comando shell a seguir criará o banco de dados `zabbix` (último parâmetro) com o usuário criado anteriormente como proprietário (`-O zabbix`).

```
sudo -u postgres createdb -O zabbix -E Unicode -T template0 zabbix
```

- Importe o `schema` e os dados iniciais (supondo que você esteja no diretório raiz das fontes do Zabbix).

Para um banco de dados de proxy do Zabbix, apenas o `schema.sql` deve ser importado (sem `images.sql` nem `data.sql`).

```
cd database/postgresql
cat schema.sql | sudo -u zabbix psql zabbix
#### stop here if you are creating database for Zabbix proxy
cat images.sql | sudo -u zabbix psql zabbix
cat data.sql | sudo -u zabbix psql zabbix
```

Attention:

Os comandos acima são fornecidos como um exemplo que funcionará na maioria das instalações em GNU/Linux. Dependendo de como seu sistema/banco de dados estiver configurado, você pode usar diferentes comandos, como:
`

psql -U <username>

` Se você tiver dificuldades ao configurar o banco de dados, por favor consulte seu administrador de banco de dados.

TimescaleDB

Instruções para criação e configuração do TimescaleDB são fornecidas em uma [seção](#) separada.

Oracle

Instruções para criação e configuração de banco de dados Oracle são fornecidas em uma [seção](#) separada.

SQLite

O uso de SQLite é suportado apenas para **Zabbix Proxy**!

O banco de dados será criado automaticamente caso não existir.

Retorne para a [seção de instalação](#).

2 Reparando o conjunto de caracteres e agrupamento do banco de dados Zabbix

MySQL/MariaDB

Historicamente, o MySQL e seus derivados usavam 'utf8' como um alias para utf8mb3 - a própria implementação de 3 bytes do UTF8 padrão do MySQL, que é de 4 bytes. A partir do MySQL 8.0.28 e MariaDB 10.6.1, o conjunto de caracteres 'utf8mb3' é obsoleto e em algum momento seu suporte será descartado enquanto 'utf8' tornar-se uma referência a 'utf8mb4'. Desde o Zabbix 6.0, 'utf8mb4' é suportado. Para evitar problemas futuros, é altamente recomendado usar 'utf8mb4'. Outra vantagem de mudar para 'utf8mb4' é o suporte de Unicode suplementar personagens.

1. Verifique o conjunto de caracteres e o agrupamento do banco de dados.

Por exemplo:

```
mysql> SELECT @@character_set_database, @@collation_database;
+-----+-----+
| @@character_set_database | @@collation_database |
+-----+-----+
| latin2 | latin2_general_ci |
+-----+-----+
```

Ou:

```
mysql> SELECT @@character_set_database, @@collation_database;
+-----+-----+
| @@character_set_database | @@collation_database |
+-----+-----+
| utf8 | utf8_bin |
+-----+-----+
```

Como vemos, o conjunto de caracteres aqui não é 'utf8mb4' e o agrupamento não é 'utf8mb4_bin', então precisamos corrigi-los.

2. Pare o Zabbix.

3. Crie uma cópia de backup do banco de dados!

4. Corrija o conjunto de caracteres e o agrupamento no nível do banco de dados:

```
alter database <your DB name> conjunto de caracteres utf8mb4 collate utf8mb4_bin;
```

Valores fixos:

```
mysql> SELECT @@character_set_database, @@collation_database;
+-----+-----+
| @@character_set_database | @@collation_database |
+-----+-----+
| utf8mb4 | utf8mb4_bin |
+-----+-----+
```

5. Carregue o [script](#) para corrigir o conjunto de caracteres e o agrupamento no nível da tabela e da coluna:

```
mysql <nome do seu banco de dados> <utf8mb4_convert.sql
```

6. Execute o script:

```
SET @ZABBIX_DATABASE = '<nome do seu banco de dados>';
Se MariaDB → definir innodb_strict_mode = OFF;
CALL zbx_convert_utf8();
Se MariaDB → definir innodb_strict_mode = ON;
procedimento de descarte zbx_convert_utf8;
```

Observe que 'utf8mb4' deve consumir um pouco mais de espaço em disco.

7. Se não houver erros - talvez você queira criar uma cópia de backup do banco de dados com o banco de dados fixo.

8. Inicie o Zabbix.

3 Atualização de banco de dados para chaves primárias

Visão geral

Desde o Zabbix 6.0, chaves primárias são usadas para todas as tabelas em novas instalações.

Esta seção fornece instruções para atualizar manualmente o tabelas de histórico em instalações existentes para chaves primárias.

As instruções estão disponíveis para:

- [MySQL](#)
- [PostgreSQL](#)
- [TimescaleDB v1](#)
- [TimescaleDB v2](#)
- [Oráculo](#)

Attention:

The instructions provided on this page are designed for advanced users. Note that these instructions might need to be adjusted for your specific configuration.

Anotações importantes

- Certifique-se de fazer backup do banco de dados antes da atualização
- Se seu banco de dados usa partições, entre em contato com o administrador do banco de dados ou com a equipe de suporte do Zabbix para obter ajuda
- Os arquivos CSV podem ser removidos após uma atualização bem-sucedida para as chaves primárias

MySQL

A exportação e importação devem ser realizadas no tmux/screen, para que a sessão não seja descartada.

Veja também: [Notas importantes](#)

MySQL 5.7+/8.0+

- Renomeie tabelas antigas, crie novas tabelas executando `history_pk_prepare.sql`.

```
mysql -uzabbix -p<senha> zabbix < /usr/share/doc/zabbix-sql-scripts/mysql/history_pk_prepare.sql
```

- Exportar e importar dados

mysqlsh deve ser [instalado](#). mysqlsh deve ser capaz de se conectar ao banco de dados. Se a conexão for feita através do socket, pode ser necessário indicar explicitamente um caminho para ele.

Conecte-se via mysqlsh:

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Executar (CSVPATH deve/pode ser ajustado de acordo com as necessidades):

```
CSVPATH="/var/lib/mysql-files";
```

```
util.exportTable("history_old", CSVPATH + "/history.csv", { dialeto: "csv" });
util.importTable(CSVPATH + "/history.csv", {"dialect": "csv", "table": "history" });
```

```
util.exportTable("history_uint_old", CSVPATH + "/history_uint.csv", { dialeto: "csv" });
util.importTable(CSVPATH + "/history_uint.csv", {"dialect": "csv", "table": "history_uint" });
```

```
util.exportTable("history_str_old", CSVPATH + "/history_str.csv", { dialeto: "csv" });
util.importTable(CSVPATH + "/history_str.csv", {"dialect": "csv", "table": "history_str" });
```

```
util.exportTable("history_log_old", CSVPATH + "/history_log.csv", { dialeto: "csv" });
util.importTable(CSVPATH + "/history_log.csv", {"dialect": "csv", "table": "history_log" });
```

```
util.exportTable("history_text_old", CSVPATH + "/history_text.csv", { dialeto: "csv" });
util.importTable(CSVPATH + "/history_text.csv", {"dialect": "csv", "table": "history_text" });
```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```

DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;

```

MySQL <5.7, MariaDB (ou se mysqlsh não puder ser usado por algum motivo)

Esta opção é mais lenta e demorada, use somente se há uma razão para não usar o mysqlsh.

- Renomeie tabelas antigas, crie novas tabelas executando `history_pk_prepare.sql`.

```
mysql -uzabbix -p<senha> zabbix < /usr/share/doc/zabbix-sql-scripts/mysql/history_pk_prepare.sql
```

- Exportar e importar dados

Verifique se a importação/exportação está habilitada apenas para arquivos no caminho específico:

```

mysql> SELECT @@secure_file_priv;
+-----+
| @@secure_file_priv |
+-----+
| /var/lib/mysql-files/ |
+-----+

```

Se o valor for um caminho para o diretório, a exportação/importação poderá ser executada para arquivos nesse diretório. Neste caso, caminhos para arquivos em consultas devem ser editado em conformidade. Alternativamente, `secure_file_priv` pode ser desabilitado (definido como string vazia) durante a atualização. Se o valor estiver vazio, exportar/importar pode ser executado de/para arquivos que podem estar localizados em qualquer lugar.

`max_execution_time` deve ser desabilitado antes de exportar os dados para evitar o tempo limite durante a exportação.

```
SET @@max_execution_time=0;
```

```

SELECT * INTO OUTFILE '/var/lib/mysql-files/history.csv' CAMPOS TERMINADOS POR ',' ESCAPED POR '"' LINHAS
LOAD DATA INFILE '/var/lib/mysql-files/history.csv' IGNORE INTO TABLE history CAMPOS TERMINADOS POR ',' ES

```

```

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_uint.csv' CAMPOS TERMINADOS POR ',' ESCAPED POR '"' LI
LOAD DATA INFILE '/var/lib/mysql-files/history_uint.csv' IGNORE INTO TABLE history_uint CAMPOS TERMINADOS

```

```

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_str.csv' CAMPOS TERMINADOS POR ',' ESCAPED POR '"' LIN
LOAD DATA INFILE '/var/lib/mysql-files/history_str.csv' IGNORE INTO TABLE history_str CAMPOS TERMINADOS PO

```

```

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_log.csv' CAMPOS TERMINADOS POR ',' ESCAPED POR '"' LIN
LOAD DATA INFILE '/var/lib/mysql-files/history_log.csv' IGNORE INTO TABLE history_log CAMPOS TERMINADOS PO

```

```

SELECT * INTO OUTFILE '/var/lib/mysql-files/history_text.csv' CAMPOS TERMINADOS POR ',' ESCAPED POR '"' LI
LOAD DATA INFILE '/var/lib/mysql-files/history_text.csv' IGNORE INTO TABLE history_text CAMPOS TERMINADOS

```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```

DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;

```

Melhorando a performance

Dicas adicionais para melhorar o desempenho em ambos os casos:

- Aumente o `bulk_insert_buffer_size` na configuração arquivo dentro da seção `[mysqld]` ou defina-o antes de importar com SET:

```

[mysqld]
bulk_insert_buffer_size=256M

```

```

mysql cli > SET SESSION bulk_insert_buffer_size= 1024 * 1024 * 256;
mysql cli > ... importar consultas ...

```

- Consulte "Otimizando o carregamento de dados em massa do InnoDB": ([MySQL 5.7](#), [MySQL 8.0](#))
- Desabilite o log binário (não deve ser usado no caso de servidores escravos, uma vez que não irá replicar dados):

```
mysql cli > SET SESSION SQL_LOG_BIN=0;
mysql cli > ... importar consultas ...
```

MySQL 8.0+ with mysqlsh

This method can be used with a running Zabbix server, but it is recommended to stop the server for the time of the upgrade. The MySQL Shell (mysqlsh) must be [installed](#) and able to connect to the DB.

- Log in to MySQL console as root (recommended) or as any user with FILE privileges.
- Start MySQL with [local_infile](#) variable enabled.
- Rename old tables and create new tables by running `history_pk_prepare.sql`.

```
mysql -uzabbix -p<password> zabbix < /usr/share/doc/zabbix-sql-scripts/mysql/history_pk_prepare.sql
```

- Export and import data.

Connect via mysqlsh. If using a socket connection, specifying the path might be required.

```
sudo mysqlsh -uroot -S /run/mysqld/mysqld.sock --no-password -Dzabbix
```

Run (CSVPATH can be changed as needed):

```
CSVPATH="/var/lib/mysql-files";

util.exportTable("history_old", CSVPATH + "/history.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history.csv", {"dialect": "csv", "table": "history" });

util.exportTable("history_uint_old", CSVPATH + "/history_uint.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_uint.csv", {"dialect": "csv", "table": "history_uint" });

util.exportTable("history_str_old", CSVPATH + "/history_str.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_str.csv", {"dialect": "csv", "table": "history_str" });

util.exportTable("history_log_old", CSVPATH + "/history_log.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_log.csv", {"dialect": "csv", "table": "history_log" });

util.exportTable("history_text_old", CSVPATH + "/history_text.csv", { dialect: "csv" });
util.importTable(CSVPATH + "/history_text.csv", {"dialect": "csv", "table": "history_text" });
```

- Follow [post-migration instructions](#) to drop the old tables.

MariaDB/MySQL 8.0+ without mysqlsh

This upgrade method takes more time and should be used only if an upgrade with mysqlsh is not possible.

Table upgrade

- Log in to MySQL console as root (recommended) or any user with FILE privileges.
- Start MySQL with [local_infile](#) variable enabled.
- Rename old tables and create new tables by running `history_pk_prepare.sql`:

```
mysql -uzabbix -p<password> zabbix < /usr/share/doc/zabbix-sql-scripts/mysql/history_pk_prepare.sql
```

Migration with stopped server

`max_execution_time` must be disabled before migrating data to avoid timeout during migration.

```
SET @@max_execution_time=0;

INSERT IGNORE INTO history SELECT * FROM history_old;
INSERT IGNORE INTO history_uint SELECT * FROM history_uint_old;
INSERT IGNORE INTO history_str SELECT * FROM history_str_old;
INSERT IGNORE INTO history_log SELECT * FROM history_log_old;
INSERT IGNORE INTO history_text SELECT * FROM history_text_old;
```

Follow [post-migration instructions](#) to drop the old tables.

Migration with running server

Check for which paths import/export is enabled:

```
mysql> SELECT @@secure_file_priv;
+-----+
| @@secure_file_priv |
+-----+
| /var/lib/mysql-files/ |
+-----+
```

If `secure_file_priv` value is a path to a directory, export/import will be performed for files in that directory. In this case, edit paths to files in queries accordingly or set the `secure_file_priv` value to an empty string for the upgrade time.

If `secure_file_priv` value is empty, export/import can be performed from any location.

If `secure_file_priv` value is NULL, set it to the path that contains exported table data ('/var/lib/mysql-files/' in the example above).

For more information, see [MySQL documentation](#).

`max_execution_time` must be disabled before exporting data to avoid timeout during export.

```
SET @@max_execution_time=0;
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TERMINATED BY '\n'
LOAD DATA INFILE '/var/lib/mysql-files/history.csv' IGNORE INTO TABLE history FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history_uint.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TERMINATED BY '\n'
LOAD DATA INFILE '/var/lib/mysql-files/history_uint.csv' IGNORE INTO TABLE history_uint FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history_str.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TERMINATED BY '\n'
LOAD DATA INFILE '/var/lib/mysql-files/history_str.csv' IGNORE INTO TABLE history_str FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history_log.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TERMINATED BY '\n'
LOAD DATA INFILE '/var/lib/mysql-files/history_log.csv' IGNORE INTO TABLE history_log FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
```

```
SELECT * INTO OUTFILE '/var/lib/mysql-files/history_text.csv' FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES TERMINATED BY '\n'
LOAD DATA INFILE '/var/lib/mysql-files/history_text.csv' IGNORE INTO TABLE history_text FIELDS TERMINATED BY ',' ESCAPED BY '"' LINES
```

Follow [post-migration instructions](#) to drop the old tables.

PostgreSQL

A exportação e importação devem ser realizadas no tmux/screen, para que a sessão não seja descartada.

Veja também: [Notas importantes](#)

Atualizando tabelas

- Renomeie tabelas usando `history_pk_prepare.sql`.

```
sudo -u zabbix psql zabbix < /usr/share/doc/zabbix-sql-scripts/postgresql/history_pk_prepare.sql
```

- Exporte o histórico atual, importe-o para a tabela temporária, e insira-o em novas tabelas enquanto ignora duplicatas

```
\copy history_old TO '/tmp/history.csv' DELIMITER ',' CSV
CRIAR TABELA TEMP temp_history (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor PRECISÃO DUPLA PADRÃO '0.0000' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
\copy temp_history FROM '/tmp/history.csv' DELIMITER ',' CSV
INSERT INTO history SELECT * FROM temp_history ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;
```

```
\copy history_uint_old TO '/tmp/history_uint.csv' DELIMITER ',' CSV
CRIAR TABELA TEMP temp_history_uint (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor numérico(20) PADRÃO '0' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
```

```

);
\copy temp_history_uint FROM '/tmp/history_uint.csv' DELIMITER ',' CSV
INSERT INTO history_uint SELECT * FROM temp_history_uint ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

\copy history_str_old TO '/tmp/history_str.csv' DELIMITER ',' CSV
CRIAR TABELA TEMP temp_history_str (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor varchar(255) DEFAULT '' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
\copy temp_history_str FROM '/tmp/history_str.csv' DELIMITER ',' CSV
INSERT INTO history_str (itemid,clock,value,ns) SELECT * FROM temp_history_str ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

\copy history_log_old TO '/tmp/history_log.csv' DELIMITER ',' CSV
CRIAR TABELA TEMP temp_history_log (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
timestamp integer DEFAULT '0' NOT NULL,
fonte varchar(64) DEFAULT '' NÃO NULO,
severidade integer DEFAULT '0' NOT NULL,
valor texto DEFAULT '' NÃO NULO,
logeventid integer DEFAULT '0' NOT NULL,
ns integer DEFAULT '0' NOT NULL
);
\copy temp_history_log FROM '/tmp/history_log.csv' DELIMITER ',' CSV
INSERT INTO history_log SELECT * FROM temp_history_log ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

\copy history_text_old TO '/tmp/history_text.csv' DELIMITER ',' CSV
CRIAR TABELA TEMP temp_history_text (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor texto DEFAULT '' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
\copy temp_history_text FROM '/tmp/history_text.csv' DELIMITER ',' CSV
INSERT INTO history_text SELECT * FROM temp_history_text ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```

DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;

```

Considere usar as seguintes dicas para melhorar o desempenho da pastilha:

- [PostgreSQL: carregamento em massa de grandes quantidades de dados](#)
- [Distância do ponto de verificação e quantidade de WAL](#)

Migration with running server

- Export current history, import it to the temp table, then insert the data into new tables while ignoring duplicates:

```

\copy history_old TO '/tmp/history.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history (
    itemid          bigint          NOT NULL,
    clock           integer         DEFAULT '0'      NOT NULL,
    value           DOUBLE PRECISION DEFAULT '0.0000' NOT NULL,
    ns              integer         DEFAULT '0'      NOT NULL
);
\copy temp_history FROM '/tmp/history.csv' DELIMITER ',' CSV
INSERT INTO history SELECT * FROM temp_history ON CONFLICT (itemid,clock,ns) DO NOTHING;

```

```

\copy history_uint_old TO '/tmp/history_uint.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_uint (
    itemid          bigint          NOT NULL,
    clock           integer         DEFAULT '0' NOT NULL,
    value           numeric(20)     DEFAULT '0' NOT NULL,
    ns              integer         DEFAULT '0' NOT NULL
);
\copy temp_history_uint FROM '/tmp/history_uint.csv' DELIMITER ',' CSV
INSERT INTO history_uint SELECT * FROM temp_history_uint ON CONFLICT (itemid, clock, ns) DO NOTHING;

\copy history_str_old TO '/tmp/history_str.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_str (
    itemid          bigint          NOT NULL,
    clock           integer         DEFAULT '0' NOT NULL,
    value           varchar(255)    DEFAULT '' NOT NULL,
    ns              integer         DEFAULT '0' NOT NULL
);
\copy temp_history_str FROM '/tmp/history_str.csv' DELIMITER ',' CSV
INSERT INTO history_str (itemid, clock, value, ns) SELECT * FROM temp_history_str ON CONFLICT (itemid, clock, ns) DO NOTHING;

\copy history_log_old TO '/tmp/history_log.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_log (
    itemid          bigint          NOT NULL,
    clock           integer         DEFAULT '0' NOT NULL,
    timestamp       integer         DEFAULT '0' NOT NULL,
    source          varchar(64)     DEFAULT '' NOT NULL,
    severity        integer         DEFAULT '0' NOT NULL,
    value           text            DEFAULT '' NOT NULL,
    logeventid      integer         DEFAULT '0' NOT NULL,
    ns              integer         DEFAULT '0' NOT NULL
);
\copy temp_history_log FROM '/tmp/history_log.csv' DELIMITER ',' CSV
INSERT INTO history_log SELECT * FROM temp_history_log ON CONFLICT (itemid, clock, ns) DO NOTHING;

\copy history_text_old TO '/tmp/history_text.csv' DELIMITER ',' CSV
CREATE TEMP TABLE temp_history_text (
    itemid          bigint          NOT NULL,
    clock           integer         DEFAULT '0' NOT NULL,
    value           text            DEFAULT '' NOT NULL,
    ns              integer         DEFAULT '0' NOT NULL
);
\copy temp_history_text FROM '/tmp/history_text.csv' DELIMITER ',' CSV
INSERT INTO history_text SELECT * FROM temp_history_text ON CONFLICT (itemid, clock, ns) DO NOTHING;

```

- Follow [post-migration instructions](#) to drop the old tables.

TimescaleDB v1.x

A exportação e importação devem ser realizadas no tmux/screen, para que a sessão não seja descartada.

Veja também: [Notas importantes](#)

Atualizando tabelas

- Renomeie tabelas usando history_pk_prepare.sql.

```
sudo -u zabbix psql zabbix < /usr/share/doc/zabbix-sql-scripts/postgresql/history_pk_prepare.sql
```

- Exemplo de atualização para **uma** tabela:

```
-- Verifique se há espaço suficiente para permitir a exportação de dados não compactados
selecione sum(before_compression_total_bytes)/1024/1024 como before_compression_total_mbytes, sum(after_compression_total_bytes)/1024/1024 como after_compression_total_mbytes
```

```
-- Exportar dados
```

```
\copy (selecione * from history_uint_old) TO '/tmp/history_uint.csv' DELIMITER ',' CSV
```

```
CRIAR TABELA TEMP temp_history_uint (
```

```

itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor numérico(20) PADRÃO '0' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
-- Importar dados
\copy temp_history_uint FROM '/tmp/history_uint.csv' DELIMITER ',' CSV

-- Criar hipertabela e preenchê-la
selecione create_hypertable('history_uint', 'clock', chunk_time_interval => 86400, migrate_data => true);
INSERT INTO history_uint SELECT * FROM temp_history_uint ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

-- Ativar a compactação
selecione set_integer_now_func('history_uint', 'zbx_ts_unix_now', true);
alter table history_uint set (timescaledb.compress,timescaledb.compress_segmentby='itemid',timescaledb.compr

-- O ID do trabalho será retornado, deve ser passado para run_job
selecione add_compress_chunks_policy('history_uint', (
selecione (p.older_than).integer_interval de _timescaledb_config.bgw_policy_compress_chunks p
junção interna _timescaledb_catalog.hypertable h on (h.id=p.hypertable_id) onde h.table_name='history_uint
)::inteiro
);

select alter_job((selecione job_id de timescaledb_information.jobs onde hypertable_schema='public' e hyper

-- Executar trabalho de compactação
chame run_job(<JOB_ID>);
-- Pode mostrar 'NOTICE: nenhum chunks for hypertable public.history_uint que satisfaça a política de comp

    • Verifique se tudo funciona como esperado

    • Derrube tabelas antigas

DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;

Veja também: Dicas para melhorar o desempenho de inserção do PostgreSQL

TimescaleDB v2.x

A exportação e importação devem ser realizadas no tmux/screen, para que a sessão não seja descartada.

Veja também: Notas importantes

Atualizando tabelas

    • Renomeie tabelas usando history_pk_prepare.sql.

sudo -u zabbix psql zabbix < /usr/share/doc/zabbix-sql-scripts/postgresql/history_pk_prepare.sql

    • Exemplo de atualização para uma tabela:

-- Verifique se há espaço suficiente para permitir a exportação de dados não compactados
selecione sum(before_compression_total_bytes)/1024/1024 como before_compression_total_mbytes, sum(after_co

-- Exportar dados
\copy (selecione * from history_uint_old) TO '/tmp/history_uint.csv' DELIMITER ',' CSV

CRIAR TABELA TEMP temp_history_uint (
itemid bigint NÃO NULO,
clock integer DEFAULT '0' NOT NULL,
valor numérico(20) PADRÃO '0' NÃO NULO,
ns integer DEFAULT '0' NOT NULL
);
-- Importar dados
\copy temp_history_uint FROM '/tmp/history_uint.csv' DELIMITER ',' CSV

```



```
-- Criar hipertabela e preenchê-la
selecione create_hypertable('history_uint', 'clock', chunk_time_interval => 86400, migrate_data => true);
INSERT INTO history_uint SELECT * FROM temp_history_uint ON CONFLICT (itemid,clock,ns) NÃO FAÇA NADA;

-- Ativar a compactação
selecione set_integer_now_func('history_uint', 'zbx_ts_unix_now', true);
alter table history_uint set (timescaledb.compress,timescaledb.compress_segmentby='itemid',timescaledb.com

-- Substitui seu esquema em hypertable_schema
-- O ID do trabalho será retornado, deve ser passado para run_job
selecione add_compression_policy('history_uint', (
selecione extract(epoch from (config::json->>'compress_after')::interval) de timescaledb_information.jobs
)::inteiro
);
```

```
select alter_job((selecione job_id de timescaledb_information.jobs onde hypertable_schema='public' e hyper
```

```
-- Executar trabalho de compactação
chame run_job(<JOB_ID>);
-- Pode mostrar 'NOTICE: nenhum chunks for hypertable public.history_uint que satisfaça a política de comp
```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```
DROP TABLE history_old;
DROP TABLE history_uint_old;
DROP TABLE history_str_old;
DROP TABLE history_log_old;
DROP TABLE history_text_old;
```

Veja também: Dicas para [melhorar o desempenho de inserção do PostgreSQL](#)

Batch migration of history tables

- Prepare directories for Data Pump.

Data Pump must have read and write permissions to these directories.

Example:

```
mkdir -pv /export/history
chown -R oracle:oracle /export
```

- Create a directory object and grant read and write permissions to this object to the user used for Zabbix authentication ('zabbix' in the example below). Under sysdba role, run:

```
create directory history as '/export/history';
grant read,write on directory history to zabbix;
```

- Export tables. Replace N with the desired thread count.

```
expdp zabbix/password@oracle_host/service \
  DIRECTORY=history \
  TABLES=history_old,history_uint_old,history_str_old,history_log_old,history_text_old \
  PARALLEL=N
```

- Import tables. Replace N with the desired thread count.

```
impdp zabbix/password@oracle_host/service \
  DIRECTORY=history \
  TABLES=history_uint_old \
  REMAP_TABLE=history_old:history,history_uint_old:history_uint,history_str_old:history_str,history_log_old:
  data_options=SKIP_CONSTRAINT_ERRORS table_exists_action=APPEND PARALLEL=N CONTENT=data_only
```

- Follow [post-migration instructions](#) to drop the old tables.

Oracle

A exportação e importação devem ser realizadas no tmux/screen, para que a sessão não seja descartada.

Veja também: [Notas importantes](#)

Importar/exportar tabelas de histórico em uma tentativa

- Instale o Oracle Data Pump ([Instant Client Tools package](#) o contém).

Além disso, considere [dicas de desempenho](#) para o Oracle Data Pump.

- Renomeie tabelas usando `history_pk_prepare.sql`.

```
shell> cd /path/to/zabbix-sources/database/oracle
```

```
shell> sqlplus zabbix/senha@oracle_host/ORCL
```

```
sqlplus> @history_pk_prepare.sql
```

- Preparar diretórios para datapump

Exemplo:

```
#### mkdir -pv /exportar/histórico
```

```
#### chown -R oracle: oracle /export
```

- Crie um objeto de diretório, conceda permissões a ele. Execute o seguinte na função sysdba:

```
crie o histórico do diretório como '/export/history';
```

```
conceder leitura, gravação no histórico do diretório para o zabbix;
```

- Exportar tabelas. Substitua N pela contagem de linhas desejada.

```
expdp zabbix/senha@172.20.0.3:1521/z \
```

```
DIRETÓRIO=histórico \
```

```
TABLES=history_old,history_uint_old,history_str_old,history_log_old,history_text_old \
```

```
PARALELO=N
```

- Importar tabelas. Substitua N pela contagem de linhas desejada.

```
impdp zabbix/senha@172.20.0.3:1521/z \
```

```
DIRETÓRIO=histórico \
```

```
TABLES=history_uint_old \
```

```
REMAP_TABLE=history_old:history,history_uint_old:history_uint,history_str_old:history_str,history_log_old
```

```
data_options=SKIP_CONSTRAINT_ERRORS table_exists_action=APPEND PARALLEL=N CONTENT=data_only
```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```
DROP TABLE history_old;
```

```
DROP TABLE history_uint_old;
```

```
DROP TABLE history_str_old;
```

```
DROP TABLE history_log_old;
```

```
DROP TABLE history_text_old;
```

Importar/exportar tabelas de histórico individualmente

- Instale o Oracle Data Pump ([Instant Client Tools package](#) o contém).

Além disso, considere [dicas de desempenho](#) para o Oracle Data Pump.

- Renomeie tabelas usando `history_pk_prepare.sql`.

```
shell> cd /path/to/zabbix-sources/database/oracle
```

```
shell> sqlplus zabbix/senha@oracle_host/ORCL
```

```
sqlplus> @history_pk_prepare.sql
```

- Preparar diretórios para datapump

Exemplo:

```
#### mkdir -pv /export/history /export/history_uint /export/history_str /export/history_log /export/histor
```

```
#### chown -R oracle: oracle /export
```

- Crie um objeto de diretório, conceda permissões a ele. Execute o seguinte na função sysdba:

```
crie o histórico do diretório como '/export/history';
```

```
conceder leitura, gravação no histórico do diretório para o zabbix;
```

```
crie o diretório history_uint como '/export/history_uint';
```

concede leitura, gravação no diretório history_uint para zabbix;

crie o diretório history_str como '/export/history_str';
conceder leitura, gravação no diretório history_str para zabbix;

crie o diretório history_log como '/export/history_log';
conceder leitura, gravação no diretório history_log para zabbix;

crie o diretório history_text como '/export/history_text';
conceder leitura, gravação no diretório history_text para zabbix;

- Exporte e importe cada tabela. Substitua N pela contagem de linhas desejada.

```
expdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=histórico TABLES=history_old PARALLEL=N
```

```
impdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history TABLES=history_old REMAP_TABLE=history_old:hist
```

```
expdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_uint TABLES=history_uint_old PARALLEL=N
```

```
impdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_uint TABLES=history_uint_old REMAP_TABLE=histor
```

```
expdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_str TABLES=history_str_old PARALLEL=N
```

```
impdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_str TABLES=history_str_old REMAP_TABLE=history_
```

```
expdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_log TABLES=history_log_old PARALLEL=N
```

```
impdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_log TABLES=history_log_old REMAP_TABLE=history_
```

```
expdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_text TABLES=history_text_old PARALLEL=N
```

```
impdp zabbix/password@172.20.0.3:1521/xe DIRECTORY=history_text TABLES=history_text_old REMAP_TABLE=histor
```

- Verifique se tudo funciona como esperado
- Derrube tabelas antigas

```
DROP TABLE history_old;  
DROP TABLE history_uint_old;  
DROP TABLE history_str_old;  
DROP TABLE history_log_old;  
DROP TABLE history_text_old;
```

Post-migration

For all databases, once the migration is completed, do the following:

- Verify that everything works as expected.
- Drop old tables:

```
DROP TABLE history_old;  
DROP TABLE history_uint_old;  
DROP TABLE history_str_old;  
DROP TABLE history_log_old;  
DROP TABLE history_text_old;
```

3 Secure connection to the database

Overview

This section provides Zabbix setup steps and configuration examples for secure TLS connections between:

Database	Zabbix components
MySQL	Zabbix frontend, Zabbix server, Zabbix proxy
PostgreSQL	Zabbix frontend, Zabbix server, Zabbix proxy

To set up connection encryption within the DBMS, see official vendor documentation for details:

- [MySQL](#): source and replica replication database servers.
- [MySQL](#): group replication, etc. database servers.
- [PostgreSQL](#) encryption options.

All examples are based on the GA releases of MySQL CE (8.0) and PgSQL (13) available through official repositories using CentOS 8.

Requirements

The following is required to set up encryption:

- Developer-supported operating system with OpenSSL $\geq 1.1.X$ or alternative.

Note:

It is recommended to avoid OS in the end-of-life status, especially in the case of new installations

- Database engine (RDBMS) installed and maintained from the official repository provided by developer. Operating systems often shipped with outdated database software versions for which encryption support is not implemented, for example RHEL 7 based systems and PostgreSQL 9.2, MariaDB 5.5 without encryption support.

Terminology

Setting this option enforces to use TLS connection to database from Zabbix server/proxy and frontend to database:

- `required` - connect using TLS as transport mode without identity checks;
- `verify_ca` - connect using TLS and verify certificate;
- `verify_full` - connect using TLS, verify certificate and verify that database identity (CN) specified by DBHost matches its certificate;

Zabbix configuration

Frontend para o banco de dados

Uma conexão segura com o banco de dados pode ser configurada durante o frontend instalação:

- Marque a caixa de seleção Database TLS encryption em **Configure DB conexão** etapa para habilitar a criptografia de transporte.
- Marque a caixa de seleção Verificar certificado de banco de dados que aparece quando O campo criptografia TLS está marcado para habilitar a criptografia com certificados.

Note:

Para MySQL, a caixa de seleção Criptografia TLS do banco de dados é desabilitado, se Database host estiver definido como localhost, porque a conexão que usa um arquivo de soquete (no Unix) ou memória compartilhada (no Windows) não pode ser criptografado.

Para PostgreSQL, a caixa de seleção criptografia TLS é desabilitada, se o valor do campo Database host começa com uma barra ou o campo está vazio.

Os seguintes parâmetros ficam disponíveis na criptografia TLS em modo de certificados (se ambas as caixas de seleção estiverem marcadas):

Parâmetro	Descrição
Arquivo de CA TLS do banco de dados	Especifique o caminho completo para um arquivo de autoridade de certificação (CA) TLS válido.
Arquivo de chave TLS do banco de dados	Especifique o caminho completo para um arquivo de chave TLS válido.
Arquivo de certificado TLS do banco de dados	Especifique o caminho completo para um arquivo de certificado TLS válido.
Verificação do host do banco de dados	Marque esta caixa de seleção para ativar a verificação do host. Desativado para MYSQL, porque a biblioteca PHP MySQL não permite pular a etapa de validação do certificado de peer.
Lista de cifras TLS do banco de dados	Especifique uma lista personalizada de cifras válidas. O formato da lista de cifras deve estar em conformidade com o padrão OpenSSL. Disponível apenas para MySQL.


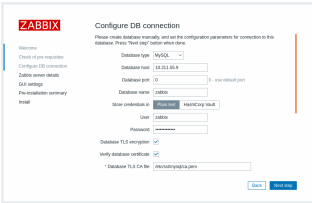
::: não importante Os parâmetros TLS devem apontar para arquivos válidos. Se eles apontar para arquivos inexistentes ou inválidos, isso levará ao erro de autorização.



Se os arquivos de certificado forem graváveis, o frontend gerará um aviso em o sistema informações[/manual/web_interface/frontend_sections informe que "os arquivos de certificado TLS devem ser somente leitura". (exibido apenas se o usuário PHP for o proprietário do certificado).

Certificados protegidos por senhas não são suportados. :::

Use cases

Zabbix frontend uses GUI interface to define possible options: required, verify_ca, verify_full. Specify required options in the installation wizard step Configure DB connections. These options are mapped to the configuration file (zabbix.conf.php) in the following manner:

GUI settings	Configuration file	Description	Result
	<pre>... // Used for TLS connection. \$DB['ENCRIPTION'] = true; \$DB['KEY_FILE'] = ""; \$DB['CERT_FILE'] = ""; \$DB['CA_FILE'] = ""; \$DB['VERIFY_HOST'] = false; \$DB['CIPHER_LIST'] = ""; ...</pre>	Check Database TLS encryption Leave Verify database certificate unchecked	Enable 'required' mode.
	<pre>... \$DB['ENCRIPTION'] = true;\\ \$DB['KEY_FILE'] = ""; \$DB['CERT_FILE'] = ""; \$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem'; \$DB['VERIFY_HOST'] = false; \$DB['CIPHER_LIST'] = ""; ...</pre>	1. Check Database TLS encryption and Verify database certificate 2. Specify path to Database TLS CA file	Enable 'verify_ca' mode.

GUI settings	Configuration file	Description	Result
	<pre> ... // Used for TLS connection with strictly defined Cipher list. \$DB['ENCRYPTION'] = true; \$DB['KEY_FILE'] = '<key_file_path>'; \$DB['CERT_FILE'] = '<key_file_path>'; \$DB['CA_FILE'] = '<key_file_path>'; \$DB['VERIFY_HOST'] = true; \$DB['CIPHER_LIST'] = '<cipher_list>'; ... Or: ... // Used for TLS connection without Cipher list defined - selected by MySQL server \$DB['ENCRYPTION'] = true; \$DB['KEY_FILE'] = '<key_file_path>'; \$DB['CERT_FILE'] = '<key_file_path>'; \$DB['CA_FILE'] = '<key_file_path>'; \$DB['VERIFY_HOST'] = true; \$DB['CIPHER_LIST'] = ''; ... </pre>	<ol style="list-style-type: none"> 1. Check Database TLS encryption and Verify database certificate 2. Specify path to Database TLS key file 3. Specify path to Database TLS CA file 4. Specify path to Database TLS certificate file 6. Specify TLS cipher list (optional) 	<p>Enable 'verify_full' mode for MySQL.</p>
	<pre> ... \$DB['ENCRYPTION'] = true; \$DB['KEY_FILE'] = '<key_file_path>'; \$DB['CERT_FILE'] = '<key_file_path>'; \$DB['CA_FILE'] = '<key_file_path>'; \$DB['VERIFY_HOST'] = true; \$DB['CIPHER_LIST'] = ''; ... </pre>	<ol style="list-style-type: none"> 1. Check Database TLS encryption and Verify database certificate 2. Specify path to Database TLS key file 3. Specify path to Database TLS CA file 4. Specify path to Database TLS certificate file 6. Check Database host verification 	<p>Enable 'verify_full' mode for PostgreSQL.</p>

See also: [Encryption configuration examples for MySQL](#), [Encryption configuration examples for PostgreSQL](#).

Zabbix server/proxy configuration

Secure connections to the database can be configured with the respective parameters in the Zabbix **server** and/or **proxy** configuration file.

Configuration	Result
None	Connection to the database without encryption.
1. Set DBTLSConnect=required	Server/proxy make a TLS connection to the database. An unencrypted connection is not allowed.

Configuration	Result
1. Set DBTLSConnect=verify_ca 2. Set DBTLSCAFile - specify the TLS certificate authority file	Server/proxy make a TLS connection to the database after verifying the database certificate.
1. Set DBTLSConnect=verify_full 2. Set DBTLSCAFile - specify TLS certificate authority file	Server/proxy make a TLS connection to the database after verifying the database certificate and the database host identity.
1. Set DBTLSCAFile - specify TLS certificate authority file 2. Set DBTLSCertFile - specify the client public key certificate file 3. Set DBTLSKeyFile - specify the client private key file	Server/proxy provide a client certificate while connecting to the database.
1. Set DBTLSCipher - the list of encryption ciphers that the client permits for connections using TLS protocols up to TLS 1.2	(MySQL) TLS connection is made using a cipher from the provided list.
or DBTLSCipher13 - the list of encryption ciphers that the client permits for connections using TLS 1.3 protocol	(PostgreSQL) Setting this option will be considered as an error.

1 configuração de criptografia MySQL

[comentário]: # (tags: ssl)

Overview

This section provides several encryption configuration examples for CentOS 8.2 and MySQL 8.0.21 and can be used as a quickstart guide for encrypting the connection to the database.

Attention:

If MySQL host is set to localhost, encryption options will not be available. In this case a connection between Zabbix frontend and the database uses a socket file (on Unix) or shared memory (on Windows) and cannot be encrypted.

Note:

List of encryption combinations is not limited to the ones listed on this page. There are a lot more combinations available.

Pre-requisites

Install MySQL database from the [official repository](#).

See [MySQL documentation](#) for details on how to use MySQL repo.

MySQL server is ready to accept secure connections using a self-signed certificate.

To see, which users are using an encrypted connection, run the following query (Performance Schema should be turned ON):

```
mysql> SELECT sbt.variable_value AS tls_version, t2.variable_value AS cipher, processlist_user AS user, pr
FROM performance_schema.status_by_thread AS sbt
JOIN performance_schema.threads AS t ON t.thread_id = sbt.thread_id
JOIN performance_schema.status_by_thread AS t2 ON t2.thread_id = t.thread_id
WHERE sbt.variable_name = 'Ssl_version' and t2.variable_name = 'Ssl_cipher'
ORDER BY tls_version;
```

Required mode

MySQL configuration

Modern versions of the database are ready out-of-the-box for 'required' **encryption mode**. A server-side certificate will be created after initial setup and launch.

Create users and roles for the main components:

```
mysql> CREATE USER
'zbx_srv'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>',
'zbx_web'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>'
REQUIRE SSL
PASSWORD HISTORY 5;
```

```
mysql> CREATE ROLE 'zbx_srv_role', 'zbx_web_role';
```

```
mysql> GRANT SELECT, UPDATE, DELETE, INSERT, CREATE, DROP, ALTER, INDEX, REFERENCES ON zabbix.* TO 'zbx_sr
```

```
mysql> GRANT SELECT, UPDATE, DELETE, INSERT ON zabbix.* TO 'zbx_web_role';
```

```
mysql> GRANT 'zbx_srv_role' TO 'zbx_srv'@'%';
```

```
mysql> GRANT 'zbx_web_role' TO 'zbx_web'@'%';
```

```
mysql> SET DEFAULT ROLE 'zbx_srv_role' TO 'zbx_srv'@'%';
```

```
mysql> SET DEFAULT ROLE 'zbx_web_role' TO 'zbx_web'@'%';
```

Note, that the X.509 protocol is not used to check identity, but the user is configured to use only encrypted connections. See [MySQL documentation](#) for more details about configuring users.

Run to check connection (socket connection cannot be used to test secure connections):

```
$ mysql -u zbx_srv -p -h 10.211.55.9 --ssl-mode=REQUIRED
```

Check current status and available cipher suites:

```
mysql> status
```

```
-----  
mysql Ver 8.0.21 for Linux on x86_64 (MySQL Community Server - GPL)
```

```
Connection id: 62
```

```
Current database:
```

```
Current user: zbx_srv@bdfdb.local
```

```
SSL: Cipher in use is TLS_AES_256_GCM_SHA384
```

```
mysql> SHOW SESSION STATUS LIKE 'Ssl_cipher_list'\G;
```

```
***** 1. row *****
```

```
Variable_name: Ssl_cipher_list
```

```
Value: TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:...
```

```
1 row in set (0.00 sec)
```

ERROR:

No query specified

Frontend

To enable transport-only encryption for connections between Zabbix frontend and the database:

- Check Database TLS encryption
- Leave Verify database certificate unchecked

ZABBIX

Welcome
Check of pre-requisites
Configure DB connection
Zabbix server details
GUI settings
Pre-installation summary
Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type

MySQL

Database host

10.211.55.9

Database port

0

0 - use default port

Database name

zabbix

Store credentials in

Plain text

HashiCorp Vault

User

zabbix

Password

Database TLS encryption

☒

Verify database certificate

☐

Back

Next step

Server

To enable transport-only encryption for connections between server and the database, configure `/etc/zabbix/zabbix_server.conf`:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Verify CA mode

Copy required MySQL CA to the Zabbix frontend server, assign proper permissions to allow the webserver to read this file.

Note:

Verify CA mode doesn't work on SLES 12 and RHEL 7 due to older MySQL libraries.

Frontend

To enable encryption with certificate verification for connections between Zabbix frontend and the database:

- Check Database TLS encryption and Verify database certificate
- Specify path to Database TLS CA file

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

GUI settings

Pre-installation summary

Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type: MySQL

Database host: 10.211.55.9

Database port: 0 (0 - use default port)

Database name: zabbix

Store credentials in: Plain text (selected), HashiCorp Vault

User: zabbix

Password:

Database TLS encryption: ☒

Verify database certificate: ☒

* Database TLS CA file: /etc/ssl/mysql/ca.pem

Back Next step

Alternatively, this can be set in `/etc/zabbix/web/zabbix.conf.php`:

```
...
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = false;
$DB['CIPHER_LIST'] = '';
...
```

Troubleshoot user using command-line tool to check if connection is possible for required user:

```
$ mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=REQUIRED --ssl-ca=/var/lib/mysql/ca.pem
```

Server

To enable encryption with certificate verification for connections between Zabbix server and the database, configure `/etc/zabbix/zabbix_server.conf`:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/mysql/ca.pem
...
```

Verify Full mode

MySQL configuration

Set MySQL CE server configuration option (/etc/my.cnf.d/server-tls.cnf) to:

```
[mysqld]
...
# in this examples keys are located in the MySQL CE datadir directory
ssl_ca=ca.pem
ssl_cert=server-cert.pem
ssl_key=server-key.pem

require_secure_transport=ON
tls_version=TLSv1.3
...
```

Keys for the MySQL CE server and client (Zabbix frontend) should be created manually according to the MySQL CE documentation: [Creating SSL and RSA certificates and keys using MySQL](#) or [Creating SSL certificates and keys using openssl](#)

Attention:

MySQL server certificate should contain the Common Name field set to the FQDN name as Zabbix frontend will use the DNS name to communicate with the database or IP address of the database host.

Create MySQL user:

```
mysql> CREATE USER
'zbx_srv'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>',
'zbx_web'@'%' IDENTIFIED WITH mysql_native_password BY '<strong_password>'
REQUIRE X509
PASSWORD HISTORY 5;
```

Check if it is possible to log in with that user:

```
$ mysql -u zbx_web -p -h 10.211.55.9 --ssl-mode=VERIFY_IDENTITY --ssl-ca=/var/lib/mysql/ca.pem --ssl-cert=
```

Frontend

To enable encryption with full verification for connections between Zabbix frontend and the database:

- Check Database TLS encryption and Verify database certificate
- Specify path to Database TLS key file
- Specify path to Database TLS CA file
- Specify path to Database TLS certificate file

Note, that Database host verification is checked and grayed out - this step cannot be skipped for MySQL.

Warning:

Cipher list should be empty, so that frontend and server can negotiate required one from the supported by both ends.



Configure DB connection

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

Pre-installation summary

Install

Database name

User

Password

Database TLS encryption ☒

Verify database certificate ☒

* Database TLS CA file

Database TLS key file

Database TLS certificate file

Database host verification ☒

Database TLS cipher list

[Back](#)

[Next step](#)

Alternatively, this can be set in `/etc/zabbix/web/zabbix.conf.php`:

```
...
// Used for TLS connection with strictly defined Cipher list.
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = 'TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_GCM_SHA256';
...
// or
...
// Used for TLS connection without Cipher list defined - selected by MySQL server
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '/etc/ssl/mysql/client-key.pem';
$DB['CERT_FILE'] = '/etc/ssl/mysql/client-cert.pem';
$DB['CA_FILE'] = '/etc/ssl/mysql/ca.pem';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = '';
...

```

Server

To enable encryption with full verification for connections between Zabbix server and the database, configure `/etc/zabbix/zabbix_server.conf`:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_full
DBTLSCAFile=/etc/ssl/mysql/ca.pem
DBTLSCertFile=/etc/ssl/mysql/client-cert.pem
DBTLSKeyFile=/etc/ssl/mysql/client-key.pem
...

```

2 Configuração de criptografia PostgreSQL

[comentário]: # (tags: ssl)

Overview

This section provides several encryption configuration examples for CentOS 8.2 and PostgreSQL 13.

Note:

Connection between Zabbix frontend and PostgreSQL cannot be encrypted (parameters in GUI are disabled), if the value of Database host field begins with a slash or the field is empty.

Pre-requisites

Install the PostgreSQL database using the [official repository](#).

PostgreSQL is not configured to accept TLS connections out-of-the-box. Please follow instructions from PostgreSQL documentation for [certificate preparation with postgresql.conf](#) and also for [user access control](#) through `pg_hba.conf`.

By default, the PostgreSQL socket is binded to the localhost, for the network remote connections allow to listen on the real network interface.

PostgreSQL settings for all **modes** can look like this:

/var/lib/pgsql/13/data/postgresql.conf:

```
...
ssl = on
ssl_ca_file = 'root.crt'
ssl_cert_file = 'server.crt'
ssl_key_file = 'server.key'
ssl_ciphers = 'HIGH:MEDIUM:+3DES:!aNULL'
ssl_prefer_server_ciphers = on
ssl_min_protocol_version = 'TLSv1.3'
...
```

For access control adjust `/var/lib/pgsql/13/data/pg_hba.conf`:

```
...
### require
hostssl all all 0.0.0.0/0 md5

### verify CA
hostssl all all 0.0.0.0/0 md5 clientcert=verify-ca

### verify full
hostssl all all 0.0.0.0/0 md5 clientcert=verify-full
...
```

Required mode

Frontend

To enable transport-only encryption for connections between Zabbix frontend and the database:

- Check Database TLS encryption
- Leave Verify database certificate unchecked

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- GUI settings
- Pre-installation summary
- Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type
PostgreSQL

Database host
10.211.55.9

Database port
0
0 - use default port

Database name
zabbix

Database schema

Store credentials in
Plain text
HashiCorp Vault

User
zabbix

Password

Database TLS encryption
☒

Verify database certificate
☐

Back
Next step

Server

To enable transport-only encryption for connections between server and the database, configure `/etc/zabbix/zabbix_server.conf`:


```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=required
...
```

Verify CA mode

Frontend

To enable encryption with certificate authority verification for connections between Zabbix frontend and the database:

- Check Database TLS encryption and Verify database certificate
- Specify path to Database TLS key file
- Specify path to Database TLS CA file
- Specify path to Database TLS certificate file



Configure DB connection

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Database name

Database schema

User

Password

Database TLS encryption ☒

Verify database certificate ☒

* Database TLS CA file

Database TLS key file

Database TLS certificate file

Database host verification ☐

[Back](#)
[Next step](#)

Alternatively, this can be set in `/etc/zabbix/web/zabbix.conf.php`:

```
...
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
$DB['VERIFY_HOST'] = false;
$DB['CIPHER_LIST'] = '';
...
```

Server

To enable encryption with certificate verification for connections between Zabbix server and the database, configure `/etc/zabbix/zabbix_server.conf`:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_ca
DBTLSCAFile=/etc/ssl/pgsql/root.crt
...
```

Verify full mode

Frontend

To enable encryption with certificate and database host identity verification for connections between Zabbix frontend and the database:

- Check Database TLS encryption and Verify database certificate
- Specify path to Database TLS key file
- Specify path to Database TLS CA file
- Specify path to Database TLS certificate file
- Check Database host verification

- Welcome
- Check of pre-requisites
- Configure DB connection
- Zabbix server details
- Pre-installation summary
- Install

Configure DB connection

Database name

Database schema

User

Password

Database TLS encryption
☒

Verify database certificate
☒

* Database TLS CA file

Database TLS key file

Database TLS certificate file

Database host verification
☒

Back

Next step

Alternatively, this can be set in `/etc/zabbix/web/zabbix.conf.php`:

```
$DB['ENCRYPTION'] = true;
$DB['KEY_FILE'] = '';
$DB['CERT_FILE'] = '';
$DB['CA_FILE'] = '/etc/ssl/pgsql/root.crt';
$DB['VERIFY_HOST'] = true;
$DB['CIPHER_LIST'] = '';
...
```

Server

To enable encryption with certificate and database host identity verification for connections between Zabbix server and the database, configure `/etc/zabbix/zabbix_server.conf`:

```
...
DBHost=10.211.55.9
DBName=zabbix
DBUser=zbx_srv
DBPassword=<strong_password>
DBTLSConnect=verify_full
DBTLSCAFile=/etc/ssl/pgsql/root.crt
DBTLSCertFile=/etc/ssl/pgsql/client.crt
DBTLSKeyFile=/etc/ssl/pgsql/client.key
...
```

5 Configuração do TimescaleDB

Visão geral

O Zabbix suporta TimescaleDB, uma solução de banco de dados baseada em PostgreSQL de particionando automaticamente os dados em pedaços baseados em tempo para oferecer suporte mais rápido desempenho em escala.

Atualmente TimescaleDB não é suportado pelo Zabbix procurador.

As instruções nesta página podem ser usadas para criar o banco de dados TimescaleDB ou migrando de tabelas PostgreSQL existentes para TimescaleDB.

Configuração

Assumimos que a extensão TimescaleDB já foi instalada no servidor de banco de dados (veja [instalação instruções](#)).

A extensão TimescaleDB também deve ser habilitada para o banco de dados específico por executando:

```
echo "CRIAR EXTENSÃO SE NÃO EXISTIR timescaledb CASCADE;" | sudo -u postgres psql zabbix
```

A execução deste comando requer privilégios de administrador de banco de dados.

Note:

Se você usa um esquema de banco de dados diferente de 'público', você precisa adicionar uma cláusula SCHEMA ao comando acima. Por exemplo.:

```
echo "CRIAR EXTENSÃO SE NÃO EXISTE timescaledb SCHEMA yourschema CASCADE;" | sudo -u postgres psql zabbix
```

Em seguida, execute o script `timescaledb.sql` localizado em `database/postgresql`. Para novas instalações, o script deve ser executado após o O banco de dados PostgreSQL foi criado com esquema/dados iniciais (veja [criação de banco de dados](#)):

```
cat /usr/share/doc/zabbix-sql-scripts/postgresql/timescaledb.sql | sudo -u zabbix psql zabbix
```

A migração do histórico existente e dos dados de tendência pode levar muito tempo. O servidor Zabbix e o frontend devem estar inativos durante o período de migração.

O script `timescaledb.sql` define os seguintes parâmetros de manutenção:

- Substituir o período do histórico do item
- Substituir período de tendência do item

A fim de usar a arrumação particionada para o histórico e as tendências, tanto essas opções devem estar ativadas. É possível usar o particionamento TimescaleDB apenas para tendências (definindo Substituir período de tendência do item) ou apenas para histórico (Substituir o período do histórico do item).

Para PostgreSQL versão 10.2 ou superior e TimescaleDB versão 1.5 ou superior, o script `timescaledb.sql` define dois parâmetros adicionais:

- Ativar a compactação
- Comprimir registros com mais de 7 dias

Todos esses parâmetros podem ser alterados em Administração → Geral → Arrumação após a instalação.

Você pode querer executar a ferramenta `timescaledb-tune` fornecida pelo TimescaleDB para otimizar os parâmetros de configuração do PostgreSQL em seu `postgresql.conf`.

Compression can be used only if both Override item history period and Override item trend period options are enabled.

All of these parameters can be changed in Administration → General → Housekeeping after the installation.

Note:

You may want to run the `timescaledb-tune` tool provided by TimescaleDB to optimize PostgreSQL configuration parameters in your `postgresql.conf`.

Compressão do TimescaleDB

A compressão nativa do TimescaleDB é suportada a partir do Zabbix 5.0 para PostgreSQL versão 10.2 ou superior e TimescaleDB versão 1.5 ou superior para todas as tabelas Zabbix que são gerenciadas pelo TimescaleDB. Durante a atualização ou migração para o TimescaleDB, a compactação inicial das tabelas grandes pode levar muito tempo.

Observe que a compactação é compatível com a licença "timescale" da comunidade de escala de tempo e não é compatível com a licença "apache" do Apache 2.0. A partir do Zabbix 6.0.7, o Zabbix detecta se a compressão é suportada. Se não for suportado, uma mensagem de aviso será gravada no log do servidor Zabbix e os usuários não poderão habilitar a compactação no frontend.

Note:

Os usuários são incentivados a se familiarizar com a documentação de compactação do [TimescaleDB](#) antes de usar a compactação.

Observe que existem certas limitações impostas pela compactação, especificamente:

- Modificações de partes compactadas (inserts, deletes, updates) não são permitidas
- Alterações de esquema para tabelas compactadas não são permitidas.

As configurações de compactação podem ser alteradas no bloco Compressão de histórico e tendências na seção Administração → Geral → Housekeeping do Zabbix frontend.

Parâmetro	Padrão	Comentários
Enable compression	Enabled	<p>Marcar ou desmarcar a caixa de seleção não ativa/desativa a compactação imediatamente. Como a compactação é feita pelo Housekeeper, as alterações entrarão em vigor em até 2 vezes o HousekeepingFrequency (definido em <code>zabbix_server.conf</code>)</p> <p>Após desabilitar a compactação, novos fragmentos que caem no período de compactação não serão compactados. No entanto, todos os dados compactados anteriormente permanecerão compactados. Para descompactar partes compactadas anteriormente, siga as instruções na documentação do TimescaleDB.</p> <p>Ao atualizar de versões mais antigas do Zabbix com suporte a TimescaleDB, a compactação não será habilitada por padrão. Este parâmetro não pode ser inferior a 7 dias.</p>
Compress records older than	7d	<p>Devido à imutabilidade dos fragmentos compactados, todos os dados atrasados (por exemplo, dados atrasados por um proxy) que forem mais antigos que esse valor serão descartados.</p>

6 Configuração do Elasticsearch

::: não importante O suporte do Elasticsearch é experimental!

:::

Zabbix suporta o armazenamento de dados históricos por meio do Elasticsearch em vez de um banco de dados. Os usuários podem escolher o local de armazenamento para histórico dados entre um banco de dados compatível e o Elasticsearch. A configuração procedimento descrito nesta seção é aplicável ao Elasticsearch versão 7.X. Caso uma versão anterior ou posterior do Elasticsearch seja usado, algumas funcionalidades podem não funcionar como pretendido.

Se todos os dados do histórico estiverem armazenados no Elasticsearch, as tendências **não** são calculadas nem armazenadas no banco de dados. Sem tendências calculado e armazenado, o período de armazenamento do histórico pode precisar ser estendido.

Configuração

Para garantir a comunicação adequada entre todos os elementos envolvidos, certifique-se de os parâmetros do arquivo de configuração do servidor e do arquivo de configuração do frontend são configurado corretamente.

Servidor e frontend Zabbix

Rascunho do arquivo de configuração do servidor Zabbix com parâmetros a serem atualizados:

```
### Opção: HistoryStorageURL
# URL de armazenamento de histórico HTTP[S].
#
# Obrigatório: não
# Predefinição:
# HistoryStorageURL=
### Opção: HistoryStorageTypes
# Lista separada por vírgulas de tipos de valores a serem enviados para o armazenamento de histórico.
#
# Obrigatório: não
# Predefinição:
# HistoryStorageTypes=uint,dbl,str,log,texto
```

Valores de parâmetro de exemplo para preencher o arquivo de configuração do servidor Zabbix com:

```
HistoryStorageURL=http://test.elasticsearch.lan:9200
HistoryStorageTypes=str,log,texto
```

Esta configuração força o Zabbix Server a armazenar valores de histórico de tipos numéricos no banco de dados correspondente e dados de histórico textual em Pesquisa elástica.

O Elasticsearch oferece suporte aos seguintes tipos de itens:

```
uint,dbl,str,log,texto
```

Explicação do tipo de item suportado:

Tipo de valor de item	Tabela de banco de dados	Tipo de pesquisa elástica
Numérico (sem sinal)	história_uint	uint
Numérico (flutuante)	histórico	dbl
Personagem	história_str	str
Log	histórico_log	log
Texto	história_texto	texto

Rascunho do arquivo de configuração de frontend do Zabbix (conf/zabbix.conf.php) com parâmetros a serem atualizados:

```
// URL do Elasticsearch (pode ser string se o mesmo URL for usado para todos os tipos).
$HISTORY['url'] = [
    'uint' => 'http://localhost:9200',
    'texto' => 'http://localhost:9200'
];
// Tipos de valor armazenados no Elasticsearch.
$HISTORY['types'] = ['uint', 'text'];
```

Valores de parâmetro de exemplo para preencher o arquivo de configuração do frontend do Zabbix com:

```
$HISTORY['url'] = 'http://test.elasticsearch.lan:9200';
$HISTORY['types'] = ['str', 'text', 'log'];
```

Esta configuração força a armazenar o histórico de Text, Character e Log valores no Elasticsearch.

Também é necessário tornar \$HISTORY global em conf/zabbix.conf.php para certifique-se de que tudo está funcionando corretamente (consulte conf/zabbix.conf.php.example para saber como fazer isso):

```
// Arquivo de configuração da interface gráfica do Zabbix.
global $DB, $HISTÓRICO;
```

Instalando o Elasticsearch e criando mapeamento

As duas etapas finais para fazer as coisas funcionarem são a instalação do Elasticsearch em si e criando o processo de mapeamento.

Para instalar o Elasticsearch, consulte [Instalação do Elasticsearch guia](#).

Note:

Mapeamento é uma estrutura de dados no Elasticsearch (semelhante a um tabela em um banco de dados). O mapeamento para todos os tipos de dados do histórico está disponível aqui: [database/elasticsearch/elasticsearch.map](#).

A criação de mapeamento é obrigatória. Algumas funcionalidades será quebrado se o mapeamento não for criado de acordo com o instrução.

Para criar mapeamento para o tipo text, envie a seguinte solicitação para Elasticsearch:

```
curl -X PUT \
  http://your-elasticsearch.here:9200/text \
  -H 'content-type:application/json' \
  -d '{
    "settings": {
      "index": {
        "number_of_replicas": 1,
        "number_of_shards": 5
      }
    },
    "mappings": {
      "properties": {
        "itemid": {
          "type": "long"
        },
        "clock": {
          "format": "epoch_second",
          "type": "date"
        }
      },
      "value": {
```

```

        "fields": {
            "analyzed": {
                "index": true,
                "type": "text",
                "analyzer": "standard"
            }
        },
        "index": false,
        "type": "text"
    }
}
}'

```

Solicitação semelhante deve ser executada para Character e Log criação de mapeamento de valores de histórico com correção de tipo correspondente.

Note:

Para trabalhar com o Elasticsearch, consulte [Página de Requisito](#) para em formação.

Note:

Housekeeper não está excluindo nenhum dado do Elasticsearch..

Armazenando dados do histórico em vários índices baseados em data

Esta seção descreve as etapas adicionais necessárias para trabalhar com pipelines e ingerir nós.

Para começar, você deve criar modelos para índices.

O exemplo a seguir mostra uma solicitação para criar o modelo uint:

```

curl -X PUT \
  http://your-elasticsearch.here:9200/_template/uint_template \
  -H 'content-type:application/json' \
  -d'{
    "index_patterns": [
      "uint*"
    ],
    "definições": {
      "índice": {
        "number_of_replicas": 1,
        "number_of_shards": 5
      }
    },
    "mapeamentos": {
      "propriedades": {
        "item": {
          "tipo": "longo"
        },
        "relógio": {
          "format": "epoch_second",
          "tipo": "data"
        },
        "valor": {
          "tipo": "longo"
        }
      }
    }
  }
}'

```

Para criar outros modelos, o usuário deve alterar a URL (a última parte é o nome do modelo), altere o campo "index_patterns" para corresponder ao nome do índice e para definir um mapeamento válido, que pode ser obtido de banco de dados/elasticsearch/elastic

Por exemplo, o comando a seguir pode ser usado para criar um modelo para índice de texto:

```
curl -X PUT \
  http://your-elasticsearch.here:9200/_template/text_template \
  -H 'content-type:application/json' \
  -d'{
    "index_patterns": [
      "texto*"
    ],
    "definições": {
      "índice": {
        "number_of_replicas": 1,
        "number_of_shards": 5
      }
    },
    "mapeamentos": {
      "propriedades": {
        "item": {
          "tipo": "longo"
        },
        "relógio": {
          "format": "epoch_second",
          "tipo": "data"
        },
        "valor": {
          "Campos": {
            "analizado": {
              "índice": verdadeiro,
              "tipo": "texto",
              "analisador": "padrão"
            }
          },
          "índice": falso,
          "tipo": "texto"
        }
      }
    }
  }'
```

Isso é necessário para permitir que o Elasticsearch defina um mapeamento válido para índices criado automaticamente. Em seguida, é necessário criar o pipeline definição. Pipeline é algum tipo de pré-processamento de dados antes colocando dados em índices. O comando a seguir pode ser usado para criar pipeline para o índice uint:

```
curl -X PUT \
  http://your-elasticsearch.here:9200/_ingest/pipeline/uint-pipeline \
  -H 'content-type:application/json' \
  -d'{
    "description": "nomeação diária do índice uint",
    "processadores": [
      {
        "date_index_name": {
          "campo": "relógio",
          "date_formats": [
            "UNIX"
          ],
          "index_name_prefix": "uint-",
          "date_rounding": "d"
        }
      }
    ]
  }'
```

O usuário pode alterar o parâmetro de arredondamento ("date_rounding") para definir um período específico de rotação do índice. Para criar outros pipelines, o usuário deve alterar a URL (a última parte é o nome do pipeline) e altere campo "index_name_prefix" para corresponder ao nome do índice.

Veja também [Elasticsearch documentação](#).

Além disso, armazenar dados históricos em vários índices baseados em datas deve também ser habilitado no novo parâmetro na configuração do servidor Zabbix:

```
### Opção: HistoryStorageDateIndex
# Habilite o pré-processamento de valores de histórico no armazenamento de histórico para armazenar valores
# 0 - desabilitar
# 1 - habilitar
#
# Obrigatório: não
# Predefinição:
# HistoryStorageDateIndex=0
```

Solução de problemas

As etapas a seguir podem ajudá-lo a solucionar problemas com a configuração do Elasticsearch:

1. Verifique se o mapeamento está correto (solicitação GET para URL de índice necessário como `http://localhost:9200/_type`).
2. Verifique se os fragmentos não estão em estado de falha (a reinicialização do Elasticsearch deve ajudar).
3. Verifique a configuração do Elasticsearch. A configuração deve permitir o acesso do host Zabbix Frontend e do host do Zabbix Server.
4. Verifique os logs do Elasticsearch.

Se você ainda está tendo problemas com sua instalação, por favor, crie um relatório de bug com todas as informações desta lista (mapeamento, logs de erros, configuração, versão, etc.)

7 Exportação em tempo real de eventos, valores de itens, tendências

Visão geral

É possível configurar a exportação em tempo real de eventos de trigger, item valores e tendências em um formato JSON delimitado por nova linha.

A exportação é feita em arquivos, onde cada linha do arquivo de exportação é um Objeto JSON. Os mapeamentos de valor não são aplicados.

Em caso de erros (os dados não podem ser gravados no arquivo de exportação ou no arquivo de exportação não pode ser renomeado ou um novo não pode ser criado após renomeá-lo), o item de dados é descartado e nunca gravado na exportação Arquivo. Ele é escrito apenas no banco de dados Zabbix. Gravando dados no O arquivo de exportação é retomado quando o problema de gravação é resolvido.

Para obter detalhes precisos sobre quais informações são exportadas, consulte o [export protocol](#).

Observe que o host/item não pode ter metadados (grupos de hosts, nome do host, item name) se o host/item foi removido após o recebimento dos dados, mas antes dos dados exportados pelo servidor.

Configuração

A exportação em tempo real de eventos de gatilho, valores de itens e tendências é configurada especificando um diretório para os arquivos de exportação - veja o `ExportDir` parâmetro no servidor [configuração](#).

Dois outros parâmetros estão disponíveis:

- `ExportFileSize` pode ser usado para definir o tamanho máximo permitido de um arquivo de exportação individual. Quando um processo precisa gravar em um arquivo, ele verifica primeiro o tamanho do arquivo. Se exceder o tamanho configurado limite, o arquivo é renomeado anexando `.old` ao seu nome e um novo arquivo com o nome original é criado.

::: não importante Será criado um arquivo por cada processo que gravar dados (ou seja, aproximadamente 4-30 arquivos). Como o tamanho padrão por arquivo de exportação é 1G, manter arquivos de exportação grandes pode drenar o espaço em disco velozes.
:::

- `ExportType` permite especificar quais tipos de entidade (eventos, histórico, tendências) serão exportados.

8 Notas específicas de distribuição sobre a configuração do Nginx para Zabbix

RHEL

Nginx está disponível apenas em EPEL:

```
# yum -y install epel-release
```

SLES 12

No SUSE Linux Enterprise Server 12, você precisa adicionar o repositório Nginx, antes de instalar o Nginx:

```
zypper addrepo -G -t yum -c 'http://nginx.org/packages/sles/12' nginx
```

Você também precisa configurar php-fpm:

```
cp /etc/php5/fpm/php-fpm.conf{.default,}  
sed -i 's/usuário = ninguém/usuário = wwwrun/; s/grupo = ninguém/grupo = www/' /etc/php5/fpm/php-fpm.conf
```

SLES 15

No SUSE Linux Enterprise Server 15, você precisa configurar o php-fpm:

```
cp /etc/php7/fpm/php-fpm.conf{.default,}  
cp /etc/php7/fpm/php-fpm.d/www.conf{.default,}  
sed -i 's/usuário = ninguém/usuário = wwwrun/; s/grupo = ninguém/grupo = www/' /etc/php7/fpm/php-fpm.d/www
```

9 Executando o agente como root

Desde o Zabbix **5.0.0**, o arquivo de serviço systemd para o agente Zabbix nos pacotes oficiais [official packages](#) inclui explicitamente diretivas para User e Group. Ambos são definidos como zabbix.

Não é mais possível configurar em qual usuário o agente Zabbix será executado por meio do arquivo `zabbix_agentd.conf`, porque o agente ignorará essa configuração e será executado como o usuário especificado no arquivo de serviço systemd. Para executar o agente Zabbix como root, você precisa fazer as modificações descritas abaixo.

Zabbix agent

Para substituir o usuário e o grupo padrão para o Zabbix Agent, execute:

```
systemctl edit zabbix-agent
```

Em seguida, adicione o seguinte conteúdo:

```
[Service]  
User=root  
Group=root
```

Recarregue os daemons e reinicie o serviço do zabbix-agent:

```
systemctl daemon-reload  
systemctl restart zabbix-agent
```

Para **Zabbix agent** isso reabilita a funcionalidade de configurar o usuário no arquivo `zabbix_agentd.conf`. Agora, você precisa definir os parâmetros de configuração `User=root` e `AllowRoot=1` no arquivo de configuração do [configuration file](#).

Zabbix agent 2

Para substituir o usuário e o grupo padrão para o Zabbix Agent 2, execute:

```
systemctl edit zabbix-agent2
```

Em seguida, adicione o seguinte conteúdo:

```
[Service]  
User=root  
Group=root
```

Recarregue os daemons e reinicie o serviço do zabbix-agent:

```
systemctl daemon-reload  
systemctl restart zabbix-agent2
```

Para o **Zabbix agent2**, isso determina completamente o usuário no qual ele é executado. Não são necessárias modificações adicionais.

10 Agente Zabbix no Microsoft Windows

Configurando o agente

O Zabbix Agent é executado como um serviço do Windows.

Você pode executar uma instância simples do Zabbix Agent ou múltiplas instâncias em um host Windows. Uma instância simples pode utilizar o arquivo de configuração em seu local padrão: `C:\zabbix_agentd.conf` ou outra localização especificada na linha de comando. Para o caso de múltiplas instâncias, cada instância do agente deverá ter seu próprio arquivo de configuração (uma das instâncias pode usar o arquivo padrão).

Junto com o código fonte do Zabbix é fornecido um exemplo de arquivo de configuração no diretório `conf/zabbix_agentd.win.conf`.

Consulte o manual de [configuração do agente no windows](#) para opções e detalhes de configuração neste sistema operacional.

Parâmetro do nome do host

Para executar [verificações ativas](#) em um host O agente Zabbix precisa ter o nome do host definido. Além disso, o nome do host valor definido no lado do agente deve corresponder exatamente ao **"Host name"** configurado para o host na parte dianteira.

O valor do nome do host no lado do agente pode ser definido pelo **Hostname** ou parâmetro **Hostnameltem** no agente [configuration file](#) - ou os valores padrão são usados se algum desses parâmetros não for especificado.

O valor padrão para o parâmetro **Hostnameltem** é o valor retornado pela chave do agente `"system.hostname"`. Para Windows, ele retorna o resultado da função `gethostname()`, que consulta os provedores de namespace para determinar o nome do host local. Se nenhum provedor de namespace responder, o nome NetBIOS será retornado.

O valor padrão para **Hostname** é o valor retornado pelo Parâmetro **Hostnameltem**. Então, com efeito, se ambos os parâmetros são não especificado, o nome do host real será o nome NetBIOS do host; Zabbix agente usará o nome do host NetBIOS para recuperar a lista de verificações ativas do servidor Zabbix e enviar os resultados para ele.

O valor padrão para **Hostname** é o valor retornado pelo Parâmetro **Hostnameltem**. Então, com efeito, se ambos os parâmetros são não especificado, o nome do host real será o nome NetBIOS do host; Zabbix agente usará o nome do host NetBIOS para recuperar a lista de verificações ativas do servidor Zabbix e enviar os resultados para ele.

A chave `"system.hostname"` suporta dois parâmetros opcionais - `type` e `transform`.

O parâmetro `Type` determina o tipo de nome que o item deve retornar. Valores suportados:

- `netbios` (padrão) - retorna o nome do host NetBIOS que é limitado a 15 símbolos e está apenas em MAIÚSCULAS;
- `host` - diferencia maiúsculas de minúsculas, retorna o nome completo e real do host do Windows (sem um domínio);
- `shorthost` (suportado desde o Zabbix 5.4.7) - retorna parte do hostname antes do primeiro ponto. Ele retornará uma string completa se o nome não contém um ponto.

O parâmetro `Transform` é suportado desde o Zabbix 5.4.7 e permite especifique uma regra de transformação adicional para o nome do host. Compatível valores:

- `nenhum` (padrão) - use a letra maiúscula original;
- `lower` - converte o texto em minúsculas.

Então, para simplificar a configuração do arquivo `zabbix_agentd.conf` e fazer unificado, duas abordagens diferentes poderiam ser usadas.

1. deixe os parâmetros **Hostname** ou **Hostnameltem** indefinidos e O agente Zabbix usará o nome do host NetBIOS como o nome do host;
2. deixe o parâmetro **Hostname** indefinido e defina **Hostnameltem** assim:
Hostnameltem=system.hostname[host] - para o agente Zabbix usar o nome de host completo e real (diferencia maiúsculas de minúsculas) do Windows como o nome do host
Hostnameltem=system.hostname[shorthost,lower] - para Zabbix agente use apenas parte do nome do host antes do primeiro ponto, convertidos em minúsculas.

O nome do host também é usado como parte do nome do serviço do Windows, que é usado para instalar, iniciar, parar e desinstalar o serviço do Windows. Para exemplo, se o arquivo de configuração do agente Zabbix especificar `Hostname=Windows_db_server`, então o agente será instalado como um Serviço do Windows `"Zabbix Agent [Windows_db_server]"`. Portanto, ter um nome de serviço do Windows diferente para cada instância do agente Zabbix, cada instância deve usar um nome de host diferente.

Instalando o agente como um serviço Windows

Para instalar uma instância simples do agente usando o arquivo padrão de configuração `c:\zabbix_agentd.conf`:

```
zabbix_agentd.exe --install
```

Attention:

Em ambientes de 64-bit, uma versão de 64-bit do Zabbix Agent será necessária para que todas as verificações relacionadas a processos em 64-bits funcione adequadamente.

Se você quiser utilizar outro arquivo de configuração que não o padrão (`c:\zabbix_agentd.conf`), você poderá definir a sua localização durante a instalação do serviço:

```
zabbix_agentd.exe --config <your_configuration_file> --install
```

O caminho completo para o arquivo de configuração deverá ser fornecido.

Múltiplas instâncias do Zabbix Agent podem ser instaladas como serviços conforme exemplo abaixo:

```
zabbix_agentd.exe --config <configuration_file_for_instance_1> --install --multiple-agents
zabbix_agentd.exe --config <configuration_file_for_instance_2> --install --multiple-agents
...
zabbix_agentd.exe --config <configuration_file_for_instance_N> --install --multiple-agents
```

Os serviços instalados deverão estar visíveis através do painel de controle.

Iniciando o agente

Para iniciar o serviço do agente você pode usar o painel de controle ou fazê-lo a partir da linha de comando.

Iniciar uma instância simples do Zabbix Agent com arquivo padrão de configuração:

```
zabbix_agentd.exe --start
```

Iniciar uma instância simples do Zabbix Agent com arquivo de configuração específico:

```
zabbix_agentd.exe --config <your_configuration_file> --start
```

Iniciar múltiplas instâncias do Zabbix Agent com arquivo de configuração específicos:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --start --multiple-agents
```

Parando o agente

Para parar o serviço do agente você pode usar o painel de controle ou fazê-lo a partir da linha de comando.

Parar uma instância simples do Zabbix Agent com arquivo padrão de configuração:

```
zabbix_agentd.exe --stop
```

Parar uma instância simples do Zabbix Agent com arquivo de configuração específico:

```
zabbix_agentd.exe --config <your_configuration_file> --stop
```

Parar múltiplas instâncias do Zabbix Agent com arquivo de configuração específicos:

```
zabbix_agentd.exe --config <configuration_file_for_this_instance> --stop --multiple-agents
```

Desinstalar o serviço do agente

Para desinstalar uma instância simples do Zabbix Agent com arquivo padrão de configuração:

```
zabbix_agentd.exe --uninstall
```

Para desinstalar uma instância simples do Zabbix Agent com arquivo de configuração específico:

```
zabbix_agentd.exe --config <your_configuration_file> --uninstall
```

Para desinstalar múltiplas instâncias do Zabbix Agent com arquivo de configuração específicos:

```
zabbix_agentd.exe --config <configuration_file_for_instance_1> --uninstall --multiple-agents zabbix_agentd.exe --config <configuration_file_for_instance_2> --uninstall --multiple-agents ... zabbix_agentd.exe --config <configuration_file_for_instance_N> --uninstall --multiple-agents
```

Limitations

Zabbix agent for Windows does not support non-standard Windows configurations where CPUs are distributed non-uniformly across NUMA nodes. If logical CPUs are distributed non-uniformly, then CPU performance metrics may not be available for some CPUs. For example, if there are 72 logical CPUs with 2 NUMA nodes, both nodes must have 36 CPUs each.

11 Configuração SAML com Okta

Esta seção descreve como configurar o Okta para habilitar o SAML 2.0 autenticação para Zabbix.

Configuração do Okta

1. Acesse <https://okta.com> e registre-se ou faça login na sua conta.

2. Na interface web do Okta, navegue até Aplicativos → Aplicativos e pressione o botão "Adicionar aplicativo" (



Create New App

3. Pressione o botão "Criar novo aplicativo" (). Em uma janela pop-up, selecione Platform: Web, Sign on method: SAML 2.0 e pressione o botão "Criar".

4. Preencha os campos na guia Configurações gerais (a primeira guia que aparece) de acordo com suas preferências e pressione "Avançar".

5. Na guia Configure SAML, insira os valores fornecidos abaixo e, em seguida, pressione "Próximo".

- Na seção **GERAL**:

- Single sign on URL: https://<your-zabbix-url>/ui/index_sso.php?acs
A caixa de seleção Usar isso para URL de destinatário e URL de destino deve ser marcado)
- URI do público-alvo (ID da entidade SP): zabbix
Observe que esse valor será usado na declaração SAML como um identificador exclusivo do provedor de serviços (se não corresponder, a operação será rejeitada). É possível especificar um URL ou qualquer string de dados neste campo.
- Estado de retransmissão padrão:
Deixe este campo em branco; se for necessário um redirecionamento personalizado, ele pode ser adicionado no Zabbix nas configurações de Administração → Usuários.
- Preencha outros campos de acordo com suas preferências.

GENERAL

Single sign on URL ?

https://<your-zabbix-url>/ui/index_sso.php?acs

☒ Use this for Recipient URL and Destination URL

☐ Allow this app to request other SSO URLs

Audience URI (SP Entity ID) ?

zabbix

Default RelayState ?

If no value is set, a blank RelayState is sent

Name ID format ?

EmailAddress ▼

Application username ?

Email ▼

Update application username on

Create and update ▼

[Show Advanced Settings](#)

Note:

Se estiver planejando usar uma conexão criptografada, gere e certificados de criptografia públicos e, em seguida, carregue o certificado público para Ok. O formulário de upload de certificado aparece quando Assertion Encryption é definido para Criptografado (clique em Mostrar Configurações Avançadas para encontrar este parâmetro).

- Na seção **ATTRIBUTE STATEMENTS (OPTIONAL)**, adicione um atributo declaração com:
 - Nome: usrEmail
 - Formato do nome: Não especificado
 - Valor: user.email

ATTRIBUTE STATEMENTS (OPTIONAL)

[LEARN MORE](#)

Name

Name format
(optional)

Value

usrEmail

Unspecified ▼

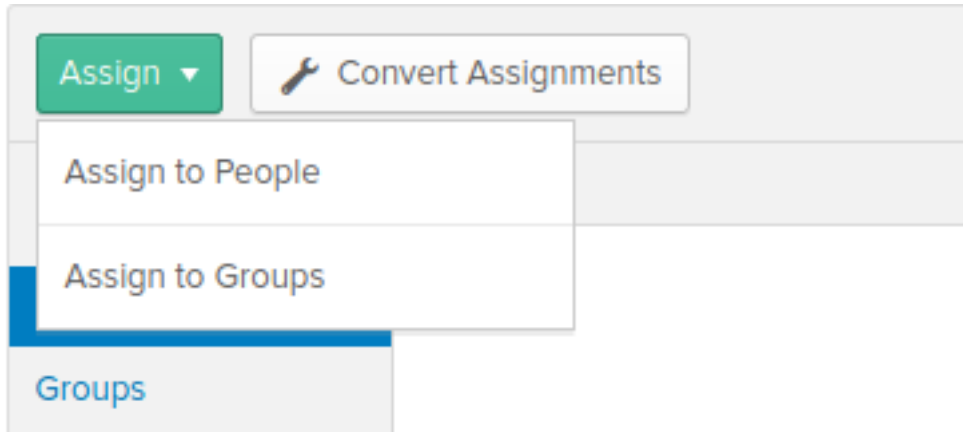
user.email ▼

Add Another

6. Na próxima guia, selecione "Sou um fornecedor de software. Gostaria de integrar meu aplicativo com Okta" e pressione

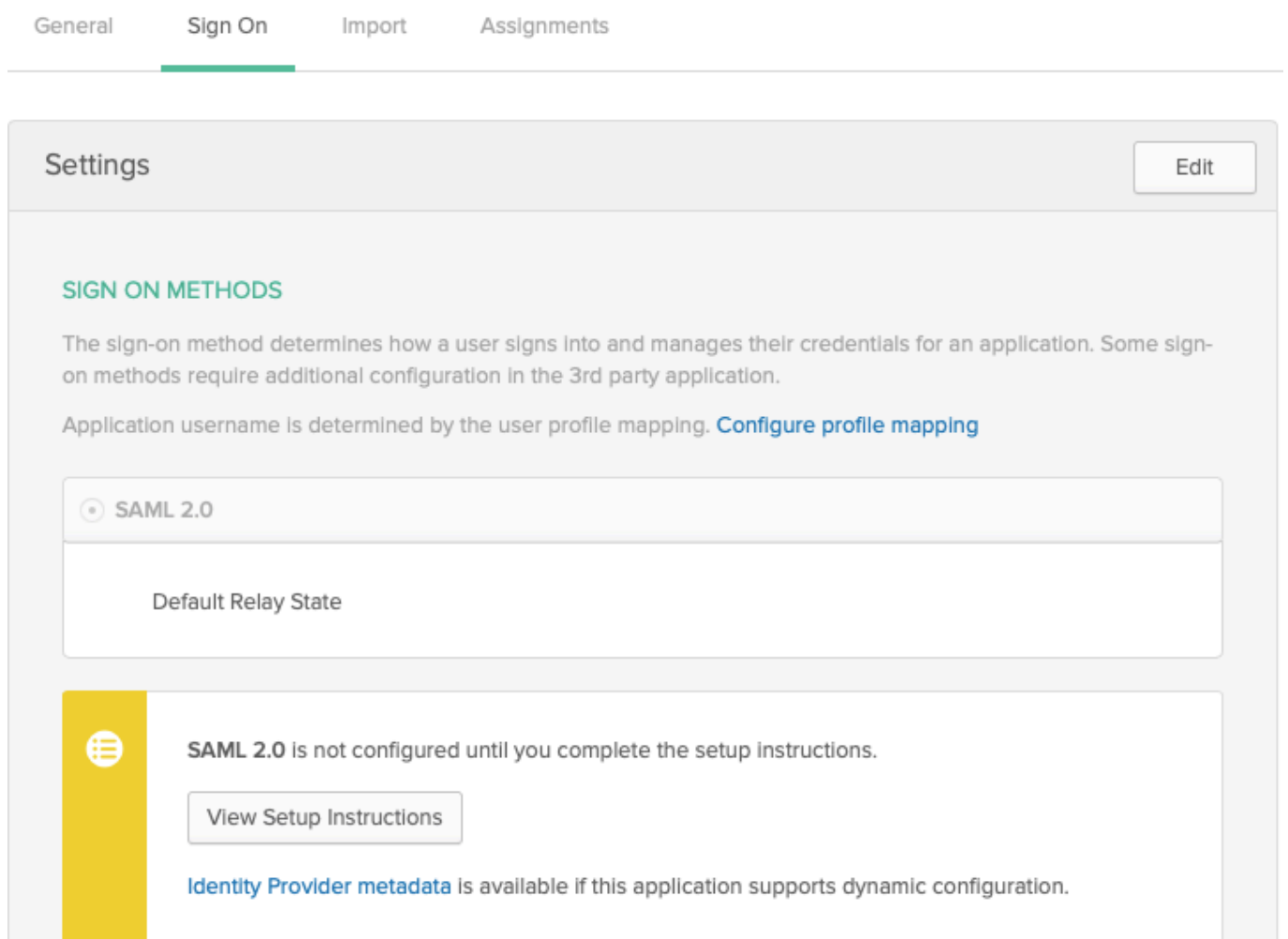
"Concluir".

7. Agora, navegue até a guia Atribuições e pressione o botão "Atribuir", em seguida, selecione Atribuir a Pessoas na lista suspensa.



8. Em um pop-up que aparece, atribua o aplicativo criado às pessoas que usarão SAML 2.0 para autenticar com o Zabbix, então pressione "Salvar e voltar".

9. Navegue até a guia Sign On e pressione o botão "View Setup Instruções". As instruções de configuração serão exibidas em uma nova guia; mantenha esta aba aberta enquanto configura o Zabbix.



Configuração do Zabbix

1. No Zabbix, vá para as configurações de SAML em Administration → Seção de autenticação e informações de cópia da configuração do Okta instruções nos campos correspondentes:

- URL de login único do provedor de identidade → URL do serviço SSO
- Emissor do provedor de identidade → ID da entidade IdP
- Atributo de nome de usuário → Nome do atributo (usrEmail)
- ID da entidade SP → URI do público-alvo

2. Baixe o certificado fornecido nas instruções de configuração do Okta page na pasta ui/conf/certs como idp.crt e defina a permissão 644 por correndo:

```
chmod 644 idp.crt
```

Observe que se você atualizou para o Zabbix 5.0 de uma versão mais antiga, você também precisará adicionar manualmente essas linhas ao arquivo zabbix.conf.php (localizado no diretório //ui/conf/ //):

```
// Usado para autenticação SAML.  
$SSO['SP_KEY'] = 'conf/certs/sp.key'; // Caminho para sua chave privada.  
$SSO['SP_CERT'] = 'conf/certs/sp.crt'; // Caminho para sua chave pública.  
$SSO['IDP_CERT'] = 'conf/certs/idp.crt'; // Caminho para a chave pública do IdP.  
$SSO['CONFIGURAÇÕES'] = []; // Configurações adicionais
```

Ver genérico [SAML Autenticação](#) instruções para obter mais detalhes.

3. Se Assertion Encryption foi definido como Criptografado no Okta, um A caixa de seleção "Assertions" do parâmetro Encrypt deve ser marcada em Zabbix também.

[Authentication](#) [HTTP settings](#) [LDAP settings](#) [SAML settings](#)

Enable SAML authentication ☒

* IdP entity ID

http://www.okta.com/xxxxxxxxxxx

* SSO service URL

https://xxx.okta.com/app/xxxxxxx_1/exkd736c1LUOFC9uY4x6/sso/saml

SLO service URL

* Username attribute

usrEmail

* SP entity ID

zabbix

SP name ID format

urn:oasis:names:tc:SAML:2.0:nameid-format:transient

Sign

☒ Messages

☒ Assertions

☐ AuthN requests

☐ Logout requests

☐ Logout responses

Encrypt

☐ Name ID

☐ Assertions

Case sensitive login

☐

Update

4. Pressione o botão "Atualizar" para salvar essas configurações.

Note:

Para entrar com SAML, o nome de usuário no Zabbix deve corresponder o e-mail Okta. Essas configurações podem ser alteradas em Administração → Seção Usuários da interface web do Zabbix.

11 Oracle database setup

Visão geral

Esta seção contém instruções para criar banco de dados Oracle e configurando conexões entre o banco de dados e o servidor Zabbix, proxy, e front-end.

Criação de banco de dados

Assumimos que existe um usuário de banco de dados zabbix com senha password e tem permissões para criar objetos de banco de dados no serviço ORCL localizado no servidor de banco de dados Oracle host. Zabbix requer um banco de dados Unicode conjunto de caracteres e um conjunto de caracteres nacional UTF8. Verifique a corrente definições:

```
sqlplus> selecione parâmetro, valor de v$nlsparms onde parâmetro='NLS_CHARACTERSET' ou parâmetro='NL
```

Agora prepare o banco de dados:

```
shell> cd /path/to/zabbix-sources/database/oracle
shell> sqlplus zabbix/senha@oracle_host/ORCL
sqlplus> @schema.sql
# pare aqui se você estiver criando um banco de dados para o proxy Zabbix
sqlplus> @imagens.sql
sqlplus> @data.sql
```

Por favor, defina o parâmetro de inicialização CURSOR_SHARING=FORCE para melhor desempenho.

Conexão configurada

O Zabbix suporta dois tipos de identificadores de conexão (métodos de conexão):

- Fácil Conexão
- Nome do serviço de rede

Parâmetros de configuração de conexão para servidor Zabbix e proxy Zabbix pode ser definido nos arquivos de configuração. Parâmetros importantes para o servidor e proxy são DBHost, DBUser, DBName e DBPassword. O mesmos parâmetros são importantes para o frontend: \$DB["SERVER"], \$DB["PORT"], \$DB["DATABASE"], \$DB["USER"], \$DB["SENHA"].

O Zabbix usa a seguinte sintaxe de string de conexão:

```
{DBUser/DBPassword[@<connect_identifier>]}
```

<connect_identifier> pode ser especificado na forma de "Net Nome do serviço" ou "Conexão fácil".

```
@[/]Host[:Porta]/<service_name> | <net_service_name>
```

Conexão fácil

O Easy Connect usa os seguintes parâmetros para se conectar ao banco de dados:

- Host - o nome do host ou endereço IP do computador do servidor de banco de dados (parâmetro DBHost no arquivo de configuração).
- Port - a porta de escuta no servidor de banco de dados (parâmetro DBPort no arquivo de configuração; se não for definido a porta padrão 1521 será usado).
- <service_name> - o nome do serviço do banco de dados que você deseja access (parâmetro DBName no arquivo de configuração).

Exemplo:

Parâmetros de banco de dados definidos no arquivo de configuração do servidor ou proxy (zabbix_server.conf e zabbix_proxy.conf):

```
DBHost=localhost
DBPort=1521
DBUser = meu nome de usuário
DBName=ORCL
DBPassword=minhasenha
```

String de conexão usada pelo Zabbix para estabelecer a conexão:

```
DBUser/DBPassword@DBHost:DBPort/DBName
```

Durante a instalação do frontend do Zabbix, defina os parâmetros correspondentes em a etapa Configure DB connection do assistente de configuração:

- Host do banco de dados: localhost
- Porta do banco de dados: 1521
- Nome do banco de dados: ORCL
- Usuário: meuusername
- Senha: minha senha



Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

GUI settings

Pre-installation summary

Install

Database type

Database host

Database port 0 - use default port

Database name

Store credentials in ☒ Plain text ☐ HashiCorp Vault

User

Password

Alternativamente, esses parâmetros podem ser definidos na configuração do frontend arquivo (zabbix.conf.php):

```
$DB["TYPE"] = 'ORACLE';
$DB["SERVER"] = 'localhost';
$DB["PORT"] = '1521';
$DB["DATABASE"] = 'ORCL';
$DB["USER"] = 'meuusuário';
$DB["SENHA"] = 'minha senha';
```

Nome do serviço de rede

Desde o Zabbix 5.4.0 é possível conectar ao Oracle usando net Nome do Serviço.

<net_service_name> é um nome simples para um serviço que resolve para um descritor de conexão.

Para usar o nome do serviço para criar uma conexão, este serviço nome deve ser definido no arquivo tnsnames.ora localizado em ambos os servidor de banco de dados e os sistemas cliente. A maneira mais fácil de ter certeza que a conexão terá sucesso é definir a localização de tnsnames.ora na variável de ambiente TNS_ADMIN. O padrão a localização do arquivo tnsnames.ora é:

```
$ORACLE_HOME/network/admin/
```

Um exemplo simples de arquivo tnsnames.ora:

```
ORCL =
(DESCRÇÃO =
  (ENDEREÇO = (PROTOCOLO = TCP)(HOST = host local)(PORT = 1521))
  (CONNECT_DATA =
    (SERVIDOR = DEDICADO)
    (SERVICE_NAME = ORCL)
  )
)
```

Para definir parâmetros de configuração para a conexão "Net Service Name" método, use uma das seguintes opções:

- Defina um parâmetro DBHost vazio e defina DBName como de costume:

```
DBHost=
DBName=ORCL
```

- Defina ambos os parâmetros e deixe ambos vazios:

```
DBHost=
DBName=
```

No segundo caso, a variável de ambiente TWO_TAKS deve ser configurada. Isto especifica o serviço Oracle remoto padrão (nome do serviço). Quando isso variável é definida, o conector se conecta ao banco de dados especificado por usando um ouvinte Oracle que aceita solicitações de conexão. Esta variável é para uso apenas em Linux e UNIX. Use a variável de ambiente LOCAL para Microsoft Windows.

Exemplo:

Conecte-se a um banco de dados usando o Net Service Name definido como ORCL e o padrão porta. Parâmetros de banco de dados definidos no arquivo de configuração do servidor ou proxy (zabbix_server.conf e zabbix_proxy.conf):

```
DBHost=  
#DBPort=  
DBUser = meu nome de usuário  
DBName=ORCL  
DBPassword=minhasenha
```

Durante a instalação do frontend do Zabbix, defina os parâmetros correspondentes em a etapa Configure DB connection do assistente de configuração:

- Hospedagem do banco de dados:
- Porta do banco de dados: 0
- Nome do banco de dados: ORCL
- Usuário: meuusername
- Senha: minha senha

Alternativamente, esses parâmetros podem ser definidos na configuração do frontend arquivo (zabbix.conf.php):

```
$DB["TYPE"] = 'ORACLE';  
$DB["SERVIDOR"] = '';  
$DB["PORT"] = '0';  
$DB["DATABASE"] = 'ORCL';  
$DB["USER"] = 'meuusuário';  
$DB["SENHA"] = 'minha senha';
```

String de conexão usada pelo Zabbix para estabelecer a conexão:

```
DBUser/DBSenha@ORCL
```

Known issues

To improve performance, you can convert the field types from nclob to nvarchar2, see [known issues](#).

12 Setting up scheduled reports

Overview

This section provides instructions on installing Zabbix web service and configuring Zabbix to enable generation of scheduled reports.

Attention:

Currently the support of scheduled reports is experimental.

Installation

A new [Zabbix web service](#) process and Google Chrome browser should be installed to enable generation of scheduled reports. The web service may be installed on the same machine where the Zabbix server is installed or on a different machine. Google Chrome browser should be installed on the same machine, where the web service is installed.

Para compilar o serviço web Zabbix a partir de fontes, veja [Instalando os serviços web do Zabbix](#).

Após a instalação, execute `zabbix_web_service` na máquina onde o web service está instalado:

```
shell> zabbix_web_service
```

Configuration

To ensure proper communication between all elements involved make sure server configuration file and frontend configuration parameters are properly configured.

Zabbix server

The following parameters in Zabbix server configuration file need to be updated: `WebServiceURL` and `StartReportWriters`.

WebServiceURL

This parameter is required to enable communication with the web service. The URL should be in the format `<host:port>/report`.

- By default, the web service listens on port 10053. A different port can be specified in the web service configuration file.
- Specifying the `/report` path is mandatory (the path is hardcoded and cannot be changed).

Example:

```
WebServiceURL=http://localhost:10053/report
```

StartReportWriters

This parameter determines how many report writer processes should be started. If it is not set or equals 0, report generation is disabled. Based on the number and frequency of reports required, it is possible to enable from 1 to 100 report writer processes.

Example:

```
StartReportWriters=3
```

Zabbix frontend

A Frontend URL parameter should be set to enable communication between Zabbix frontend and Zabbix web service:

- Proceed to the Administration → General → Other parameters frontend menu section
- Specify the full URL of the Zabbix web interface in the Frontend URL parameter.

Other configuration parameters ▾

O pacote oficial `zabbix-web-service` está disponível no [Zabbix repositório](#). O navegador Google Chrome não é incluído nesses pacotes e deve ser instalado separadamente.

14 Idiomas frontend adicionais

Visão geral

Para usar qualquer outro idioma que não o inglês na interface web do Zabbix, sua localidade deve ser instalada no servidor web. Além disso, o PHP A extensão `gettext` é necessária para que as traduções funcionem.

Instalando localidades

Para listar todos os idiomas instalados, execute:

```
localidade -a
```

Se alguns idiomas necessários não estiverem listados, abra o `/etc/locale.gen` e descomente os códigos de idioma necessários. Desde o Zabbix usa a codificação UTF-8, você precisa selecionar localidades com o conjunto de caracteres UTF-8.

Agora, execute:

locale-gen

Reinicie o servidor web.

As localidades agora devem ser instaladas. Pode ser necessário recarregar o Zabbix página de frontend no navegador usando Ctrl + F5 para que novos idiomas apareçam.

Instalando o Zabbix

Se instalar o Zabbix diretamente do [Zabbix git repositório](#), arquivos de tradução devem ser gerados manualmente. Para gerar a tradução arquivos, execute:

```
fazer gettext
locale/make_mo.sh
```

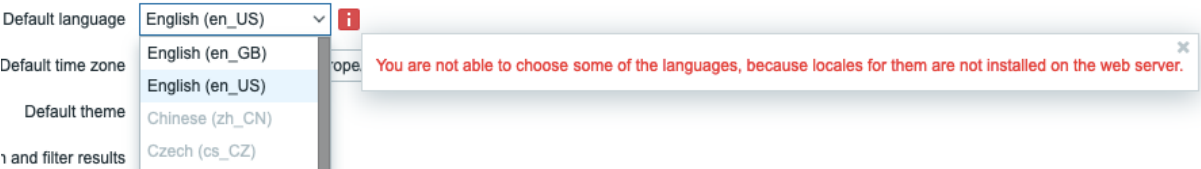
Este passo não é necessário ao instalar o Zabbix a partir de pacotes ou fontes arquivos tar.gz.

Selecionando um idioma

Existem várias maneiras de selecionar um idioma na interface web do Zabbix:

- Ao instalar a interface web - no frontend **assistente de instalação**. Selecionado idioma será definido como padrão do sistema.
- Após a instalação, o idioma padrão do sistema pode ser alterado em o Administração→Geral→GUI **menu seção**.
- O idioma de um usuário específico pode ser alterado no **perfil de usuário**.

Se uma localidade para um idioma não estiver instalada na máquina, este idioma ficará acinzentado no seletor de idioma do Zabbix. Um ícone vermelho é exibido ao lado do seletor de idioma se pelo menos uma nacionalidade for ausente. Ao pressionar este ícone a seguinte mensagem será exibido: "Você não pode escolher alguns dos idiomas, porque a nacionalidade para eles não estão instalados no servidor web."



3 Configuração do processo

1 Servidor Zabbix

Visão geral

Esta seção lista os parâmetros suportados em um arquivo de configuração do Zabbix Server (zabbix_server.conf).

Note que:

- Os valores padrão refletem os padrões do daemon, não os valores dos arquivos de configuração enviados;
- O Zabbix suporta arquivos de configuração apenas com codificação UTF-8 sem [BOM](#);
- Comentários começando com "#" são suportados apenas no início da linha.

Parameters

Parameter	Mandatory	Range	Default	Description
AlertScriptsPath	no		/usr/local/share/zabbix/alertscripts	Alert scripts (depends on compile-time installation variable datadir).
AllowRoot	no		0	Allow the server to run as 'root'. If disabled and the server is started by 'root', the server will try to switch to the 'zabbix' user instead. Has no effect if started under a regular user. 0 - do not allow 1 - allow

Parameter	Mandatory	Range	Default	Description
AllowUnsupportedDBVersions	no		0	Allow the server to work with unsupported database versions. 0 - do not allow 1 - allow
CacheSize	no	128K-64G	8M	Size of configuration cache, in bytes. Shared memory size for storing host, item and trigger data.
CacheUpdateFrequency	no	1-3600	60	How often Zabbix will perform update of configuration cache, in seconds. See also runtime control options.
DBHost	no		localhost	Database host name. In case of MySQL localhost or empty string results in using a socket. In case of PostgreSQL only empty string results in attempt to use socket.
DBName	yes			In case of Oracle empty string results in using the Net Service Name connection method; in this case consider using the TNS_ADMIN environment variable to specify the directory of the tnsnames.ora file. Database name. In case of Oracle if the Net Service Name connection method is used, specify the service name from tnsnames.ora or set to empty string; set the TWO_TASK environment variable if DBName is set to empty string.
DBPassword	no			Database password. Comment this line if no password is used.
DBPort	no	1024-65535		Database port when not using local socket. In case of Oracle if the Net Service Name connection method is used this parameter will be ignored; the port number from the tnsnames.ora file will be used instead.
DBSchema	no			Schema name. Used for PostgreSQL.
DBSocket	no			Path to MySQL socket file.
DBUser	no			Database user.

Parameter	Mandatory	Range	Default	Description
DBTLSConnect	no			<p>Setting this option enforces to use TLS connection to database:</p> <p>required - connect using TLS verify_ca - connect using TLS and verify certificate verify_full - connect using TLS, verify certificate and verify that database identity specified by DBHost matches its certificate</p> <p>On MySQL starting from 5.7.11 and PostgreSQL the following values are supported: "required", "verify_ca", "verify_full". On MariaDB starting from version 10.2.6 "required" and "verify_full" values are supported.</p> <p>By default not set to any option and the behavior depends on database configuration.</p>
DBTLSCAFile	no (yes, if DBTLSConnect set to one of: verify_ca, verify_full)			<p>This parameter is supported since Zabbix 5.0.0.</p> <p>Full pathname of a file containing the top-level CA(s) certificates for database certificate verification.</p>
DBTLSCertFile	no			<p>This parameter is supported since Zabbix 5.0.0.</p> <p>Full pathname of file containing Zabbix server certificate for authenticating to database.</p>
DBTLSKeyFile	no			<p>This parameter is supported since Zabbix 5.0.0.</p> <p>Full pathname of file containing the private key for authenticating to database.</p>
DBTLSCipher	no			<p>This parameter is supported since Zabbix 5.0.0.</p> <p>The list of encryption ciphers that Zabbix server permits for TLS protocols up through TLSv1.2.</p>
DBTLSCipher13	no			<p>Supported only for MySQL.</p> <p>This parameter is supported since Zabbix 5.0.0.</p> <p>The list of encryption ciphersuites that Zabbix server permits for TLSv1.3 protocol.</p> <p>Supported only for MySQL, starting from version 8.0.16.</p> <p>This parameter is supported since Zabbix 5.0.0.</p>

Parameter	Mandatory	Range	Default	Description
DebugLevel	no	0-5	3	Specifies debug level: 0 - basic information about starting and stopping of Zabbix processes 1 - critical information 2 - error information 3 - warnings 4 - for debugging (produces lots of information) 5 - extended debugging (produces even more information) See also runtime control options.
ExportDir	no			Directory for real-time export of events, history and trends in newline-delimited JSON format. If set, enables real-time export. This parameter is supported since Zabbix 4.0.0.
ExportFileSize	no	1M-1G	1G	Maximum size per export file in bytes. Only used for rotation if ExportDir is set. This parameter is supported since Zabbix 4.0.0.
ExportType	no			List of comma-delimited entity types (events, history, trends) for real-time export (all types by default). Valid only if ExportDir is set. Note that if ExportType is specified, but ExportDir is not, then this is a configuration error and the server will not start. e.g.: ExportType=history,trends - export history and trends only ExportType=events - export events only
ExternalScripts	no		/usr/local/share/zabbix/external_scripts	Location of external scripts (depends on compile-time installation variable datadir).
Fping6Location	no		/usr/sbin/fping6	Location of fping6. Make sure that fping6 binary has root ownership and SUID flag set. Make empty ("Fping6Location=") if your fping utility is capable to process IPv6 addresses.
FpingLocation	no		/usr/sbin/fping	Location of fping. Make sure that fping binary has root ownership and SUID flag set!

Parameter	Mandatory	Range	Default	Description
HANodeName	no			The high availability cluster node name. When empty the server is working in standalone mode and a node with empty name is created.
HistoryCacheSize	no	128K-2G	16M	Size of history cache, in bytes. Shared memory size for storing history data.
HistoryIndexCacheSize	no	128K-2G	4M	Size of history index cache, in bytes. Shared memory size for indexing history data stored in history cache. The index cache size needs roughly 100 bytes to cache one item.
HistoryStorageDateIndex	no		0	This parameter is supported since Zabbix 3.0.0. Enable preprocessing of history values in history storage to store values in different indices based on date: 0 - disable 1 - enable
HistoryStorageURL	no			History storage HTTP[S] URL. This parameter is used for Elasticsearch setup.
HistoryStorageTypes	no		uint,dbl,str,log,text	Comma separated list of value types to be sent to the history storage. This parameter is used for Elasticsearch setup.

Parameter	Mandatory	Range	Default	Description
HousekeepingFrequency	no	0-24	1	<p>How often Zabbix will perform housekeeping procedure (in hours). Housekeeping is removing outdated information from the database.</p> <p>Note: To prevent housekeeper from being overloaded (for example, when history and trend periods are greatly reduced), no more than 4 times HousekeepingFrequency hours of outdated information are deleted in one housekeeping cycle, for each item. Thus, if HousekeepingFrequency is 1, no more than 4 hours of outdated information (starting from the oldest entry) will be deleted per cycle.</p> <p>Note: To lower load on server startup housekeeping is postponed for 30 minutes after server start. Thus, if HousekeepingFrequency is 1, the very first housekeeping procedure after server start will run after 30 minutes, and will repeat with one hour delay thereafter.</p> <p>Since Zabbix 3.0.0 it is possible to disable automatic housekeeping by setting HousekeepingFrequency to 0. In this case the housekeeping procedure can only be started by housekeeper_execute runtime control option and the period of outdated information deleted in one housekeeping cycle is 4 times the period since the last housekeeping cycle, but not less than 4 hours and not greater than 4 days.</p> <p>See also runtime control options.</p>
Include	no			<p>You may include individual files or all files in a directory in the configuration file.</p> <p>To only include relevant files in the specified directory, the asterisk wildcard character is supported for pattern matching. For example:</p> <p><code>/absolute/path/to/config/files/*.</code></p> <p>See special notes about limitations.</p>

Parameter	Mandatory	Range	Default	Description
JavaGateway	no			IP address (or hostname) of Zabbix Java gateway. Only required if Java pollers are started.
JavaGatewayPort	no	1024-32767	10052	Port that Zabbix Java gateway listens on.
ListenBacklog	no	0 - INT_MAX	SOMAXCONN	The maximum number of pending connections in the TCP queue. Default value is a hard-coded constant, which depends on the system. Maximum supported value depends on the system, too high values may be silently truncated to the 'implementation-specified maximum'.
ListenIP	no		0.0.0.0	List of comma delimited IP addresses that the trapper should listen on. Trapper will listen on all network interfaces if this parameter is missing.
ListenPort	no	1024-32767	10051	Listen port for trapper.
LoadModule	no			Module to load at server startup. Modules are used to extend functionality of the server. Formats: LoadModule=<module.so> LoadModule=<path/module.so> LoadModule=</abs_path/module.so> Either the module must be located in directory specified by LoadModulePath or the path must precede the module name. If the preceding path is absolute (starts with '/') then LoadModulePath is ignored. It is allowed to include multiple LoadModule parameters.
LoadModulePath	no			Full path to location of server modules. Default depends on compilation options.
LogFile	yes, if LogType is set to file, otherwise no			Name of log file.
LogFileSize	no	0-1024	1	Maximum size of log file in MB. 0 - disable automatic log rotation. Note: If the log file size limit is reached and file rotation fails, for whatever reason, the existing log file is truncated and started anew.

Parameter	Mandatory	Range	Default	Description
LogType	no		file	Log output type: file - write log to file specified by LogFile parameter, system - write log to syslog, console - write log to standard output. This parameter is supported since Zabbix 3.0.0.
LogSlowQueries	no	0-3600000	0	How long a database query may take before being logged (in milliseconds). 0 - don't log slow queries. This option becomes enabled starting with DebugLevel=3.
MaxHousekeeperDelete	no	0-1000000	5000	No more than 'MaxHousekeeperDelete' rows (corresponding to [tablename], [field], [value]) will be deleted per one task in one housekeeping cycle. If set to 0 then no limit is used at all. In this case you must know what you are doing, so as not to overload the database! ² This parameter applies only to deleting history and trends of already deleted items.
NodeAddress	no			IP or hostname with optional port to override how the frontend should connect to the server. Format: <address>[:port] The priority of addresses used by the frontend to specify the server address is: NodeAddress (1); ListenIP (if not 0.0.0.0 or ::) (2); localhost (default) (3) See also: HANodeName parameter
PidFile	no		/tmp/zabbix_server.pid	Name of PID file.
ProxyConfigFrequency	no	1-604800	3600	How often Zabbix server sends configuration data to a Zabbix proxy in seconds. Used only for proxies in a passive mode.
ProxyDataFrequency	no	1-3600	1	How often Zabbix server requests history data from a Zabbix proxy in seconds. Used only for proxies in a passive mode.
SNMPTrapperFile	no		/tmp/zabbix_traps.tmp	Temporary file used for passing data from SNMP trap daemon to the server. Must be the same as in zabbix_trap_receiver.pl or SNMPTT configuration file.

Parameter	Mandatory	Range	Default	Description
SocketDir	no		/tmp	Directory to store IPC sockets used by internal Zabbix services. This parameter is supported since Zabbix 3.4.0.
SourceIP	no			Source IP address for: - outgoing connections to Zabbix proxy and Zabbix agent; - agentless connections (VMware, SSH, JMX, SNMP, Telnet and simple checks); - HTTP agent connections; - script item JavaScript HTTP requests; - preprocessing JavaScript HTTP requests; - sending notification emails (connections to SMTP server); - webhook notifications (JavaScript HTTP connections); - connections to the Vault
SSHKeyLocation	no			Location of public and private keys for SSH checks and actions
SSLCertLocation	no			Location of SSL client certificate files for client authentication. This parameter is used in web monitoring only.
SSLKeyLocation	no			Location of SSL private key files for client authentication. This parameter is used in web monitoring only.
SSLCALocation	no			Override the location of certificate authority (CA) files for SSL server certificate verification. If not set, system-wide directory will be used. Note that the value of this parameter will be set as libcurl option CURLOPT_CAPATH. For libcurl versions before 7.42.0, this only has effect if libcurl was compiled to use OpenSSL. For more information see CURL web page . This parameter is used in web monitoring since Zabbix 2.4.0 and in SMTP authentication since Zabbix 3.0.0.
StartAlerters	no	1-100	3	Number of pre-forked instances of alerter . This parameter is supported since Zabbix 3.4.0.

Parameter	Mandatory	Range	Default	Description
StartDBSyncers	no	1-100	4	Number of pre-forked instances of history syncers . Note: Be careful when changing this value, increasing it may do more harm than good. Roughly, the default value should be enough to handle up to 4000 NVPS.
StartDiscoverers	no	0-250	1	Number of pre-forked instances of discoverers .
StartEscalators	no	1-100	1	Number of pre-forked instances of escalators . This parameter is supported since Zabbix 3.0.0.
StartHistoryPollers	no	0-1000	5	Number of pre-forked instances of history pollers . This parameter is supported since Zabbix 5.4.0.
StartHTTPPollers	no	0-1000	1	Number of pre-forked instances of HTTP pollers ¹ .
StartIPMIPollers	no	0-1000	0	Number of pre-forked instances of IPMI pollers .
StartJavaPollers	no	0-1000	0	Number of pre-forked instances of Java pollers ¹ .
StartLLDProcessors	no	1-100	2	Number of pre-forked instances of low-level discovery (LLD) workers ¹ . The LLD manager process is automatically started when an LLD worker is started. This parameter is supported since Zabbix 4.2.0.
StartPingers	no	0-1000	1	Number of pre-forked instances of ICMP pingers ¹ .
StartPollersUnreachable	no	0-1000	1	Number of pre-forked instances of pollers for unreachable hosts (including IPMI and Java) ¹ . At least one poller for unreachable hosts must be running if regular, IPMI or Java pollers are started.
StartPollers	no	0-1000	5	Number of pre-forked instances of pollers ¹ .
StartPreprocessors	no	1-1000	3	Number of pre-forked instances of preprocessing workers ¹ . The preprocessing manager process is automatically started when a preprocessor worker is started. This parameter is supported since Zabbix 3.4.0.
StartProxyPollers	no	0-250	1	Number of pre-forked instances of pollers for passive proxies ¹ .

Parameter	Mandatory	Range	Default	Description
StartReportWriters	no	0-100	0	Number of pre-forked instances of report writers . If set to 0, scheduled report generation is disabled. The report manager process is automatically started when a report writer is started. This parameter is supported since Zabbix 5.4.0.
StartSNMPTrapper	no	0-1	0	If set to 1, SNMP trapper process will be started.
StartTimers	no	1-1000	1	Number of pre-forked instances of timers . Timers process maintenance periods.
StartTrappers	no	0-1000	5	Number of pre-forked instances of trappers ¹ . Trappers accept incoming connections from Zabbix sender, active agents and active proxies.
StartVMwareCollectors	no	0-250	0	Number of pre-forked VMware collector instances.
StatsAllowedIP	no			List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of external Zabbix instances. Stats request will be accepted only from the addresses listed here. If this parameter is not set no stats requests will be accepted. If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally and '::/0' will allow any IPv4 or IPv6 address. '0.0.0.0/0' can be used to allow any IPv4 address. Example: StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001::
Timeout	no	1-30	3	This parameter is supported since Zabbix 4.2.0. Specifies how long we wait for agent, SNMP device or external check (in seconds).
TLSCAFile	no			Full pathname of a file containing the top-level CA(s) certificates for peer certificate verification, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.

Parameter	Mandatory	Range	Default	Description
TLSCertFile	no			Full pathname of a file containing the server certificate or certificate chain, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSCipherAll	no			GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for certificate- and PSK-based encryption. Example: TLS_AES_256_GCM_SHA384:TLS_CHACHA20
TLSCipherAll13	no			This parameter is supported since Zabbix 4.4.7. Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for certificate- and PSK-based encryption. Example for GnuTLS: NONE:+VERS- TLS1.2:+ECDHE- RSA:+RSA:+ECDHE- PSK:+PSK:+AES-128- GCM:+AES-128- CBC:+AEAD:+SHA256:+SHA1:+CURVE- ALL:+COMP-NUL::+SIGN- ALL:+CTYPE-X.509 Example for OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128
TLSCipherCert	no			This parameter is supported since Zabbix 4.4.7. GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for certificate-based encryption. Example for GnuTLS: NONE:+VERS- TLS1.2:+ECDHE- RSA:+RSA:+AES-128- GCM:+AES-128- CBC:+AEAD:+SHA256:+SHA1:+CURVE- ALL:+COMP-NUL::+SIGN- ALL:+CTYPE-X.509 Example for OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128

Parameter	Mandatory	Range	Default	Description
TLSCipherCert13	no			Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for certificate-based encryption. This parameter is supported since Zabbix 4.4.7.
TLSCipherPSK	no			GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for PSK-based encryption. Example for GnuTLS: NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP=NULL:+SIGN-ALL Example for OpenSSL: kECDHEPSK+AES128:kPSK+AES128 This parameter is supported since Zabbix 4.4.7.
TLSCipherPSK13	no			Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for PSK-based encryption. Example: TLS_CHACHA20_POLY1305_SHA256:TLS_AES This parameter is supported since Zabbix 4.4.7.
TLSCRLFile	no			Full pathname of a file containing revoked certificates. This parameter is used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSKeyFile	no			Full pathname of a file containing the server private key, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TmpDir	no		/tmp	Temporary directory.
TrapperTimeout	no	1-300	300	Specifies how many seconds trapper may spend processing new data.
TrendCacheSize	no	128K-2G	4M	Size of trend cache, in bytes. Shared memory size for storing trends data.
TrendFunctionCacheSize	no	128K-2G	4M	Size of trend function cache, in bytes. Shared memory size for caching calculated trend function data.

Parameter	Mandatory	Range	Default	Description
UnavailableDelay	no	1-3600	60	How often host is checked for availability during the unavailability period, in seconds.
UnreachableDelay	no	1-3600	15	How often host is checked for availability during the unreachability period, in seconds.
UnreachablePeriod	no	1-3600	45	After how many seconds of unreachability treat a host as unavailable.
User	no		zabbix	Drop privileges to a specific, existing user on the system. Only has effect if run as 'root' and AllowRoot is disabled.
ValueCacheSize	no	0,128K-64G	8M	Size of history value cache, in bytes. Shared memory size for caching item history data requests. Setting to 0 disables value cache (not recommended). When value cache runs out of the shared memory a warning message is written to the server log every 5 minutes.
VaultDBPath	no			Vault path from where credentials for database will be retrieved by keys 'password' and 'username'. Example: secret/zabbix/database This option can only be used if DBUser and DBPassword are not specified. This parameter is supported since Zabbix 5.2.0.
VaultToken	no			Vault authentication token that should have been generated exclusively for Zabbix server with read-only permission to the paths specified in Vault macros and read-only permission to the path specified in the optional VaultDBPath configuration parameter. It is an error if VaultToken and VAULT_TOKEN environment variable are defined at the same time. This parameter is supported since Zabbix 5.2.0.
VaultURL	no		https://127.0.0.1:8200	Vault server HTTP[S] URL. System-wide CA certificates directory will be used if SSLCALocation is not specified. This parameter is supported since Zabbix 5.2.0.

Parameter	Mandatory	Range	Default	Description
VMwareCacheSize	no	256K-2G	8M	Shared memory size for storing VMware data. A VMware internal check <code>zabbix[vmware,buffer,...]</code> can be used to monitor the VMware cache usage (see Internal checks). Note that shared memory is not allocated if there are no vmware collector instances configured to start.
VMwareFrequency	no	10-86400	60	Delay in seconds between data gathering from a single VMware service. This delay should be set to the least update interval of any VMware monitoring item.
VMwarePerfFrequency	no	10-86400	60	Delay in seconds between performance counter statistics retrieval from a single VMware service. This delay should be set to the least update interval of any VMware monitoring item that uses VMware performance counters.
VMwareTimeout	no	1-300	10	The maximum number of seconds vmware collector will wait for a response from VMware service (vCenter or ESX hypervisor).
WebServiceURL	no			HTTP[S] URL to Zabbix web service in the format <code><host:port>/report</code> . For example: <code>http://localhost:10053/report</code> This parameter is supported since Zabbix 5.4.0.

Notas de rodapé

¹ Note que muitos processos de coleta de dados (pollers, pollers inalcançável, pollers ODBC, pollers HTTP, pollers JAVA, operadores de ping, trappers, proxypollers) em conjunto com gerenciador IPMI, trapper SNMP e operadores de pré-processamento podem **exaurir** o limite de descritor de arquivo por processo para o gerenciador de pré-processamento.

Warning:

Isto acarretará na parada do Zabbix Server (usualmente logo após o início, mas algumas vezes isto pode levar mais tempo). O arquivo de configuração deve ser revisado ou o limite deve ser aumentado para evitar esta situação.

² Quando uma grande quantidade de itens são eliminados isto aumenta a carga no banco de dados, porque o processo de limpeza (housekeeper) precisará remover todo o dado histórico que este itens possuíam. Por exemplo, se tivermos que remover apenas 1 protótipo de item, mas este protótipo estiver associado a 50 hosts e para cada host o protótipo for expandido para 100 itens reais, 5000 itens no total terão de ser removidos (1*50*100). Se 500 estiver configurado em `MaxHousekeeperDelete` (`MaxHousekeeperDelete=500`), o processo de limpeza terá que remover até 2500000 valores (5000*500) para os itens eliminados do histórico e tabelas de tendência em um ciclo.

2 Zabbix Proxy

Visão geral

Esta seção lista os parâmetros suportados em um arquivo de configuração do Zabbix Proxy (`zabbix_proxy.conf`). Note que:

- Os valores padrão refletem os padrões do serviço, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados somente no início da linha.

Parameters

Parameter	Mandatory	Range	Default	Description
AllowRoot	no		0	Allow the proxy to run as 'root'. If disabled and the proxy is started by 'root', the proxy will try to switch to the 'zabbix' user instead. Has no effect if started under a regular user. 0 - do not allow 1 - allow
AllowUnsupportedDBVersions	no		0	Allow the proxy to work with unsupported database versions. 0 - do not allow 1 - allow
CacheSize	no	128K-64G	8M	Size of configuration cache, in bytes.
ConfigFrequency	no	1-604800	3600	Shared memory size, for storing host and item data. How often proxy retrieves configuration data from Zabbix server in seconds. Active proxy parameter. Ignored for passive proxies (see ProxyMode parameter).
DataSenderFrequency	no	1-3600	1	Proxy will send collected data to the server every N seconds. Note that active proxy will still poll Zabbix server every second for remote command tasks. Active proxy parameter. Ignored for passive proxies (see ProxyMode parameter).
DBHost	no		localhost	Database host name. In case of MySQL localhost or empty string results in using a socket. In case of PostgreSQL only empty string results in attempt to use socket. In case of Oracle empty string results in using the Net Service Name connection method; in this case consider using the TNS_ADMIN environment variable to specify the directory of the tnsnames.ora file.

Parameter	Mandatory	Range	Default	Description
DBName	yes			<p>Database name or path to database file for SQLite3 (multi-process architecture of Zabbix does not allow to use in-memory database, e.g. <code>:memory:</code>, <code>file::memory:?cache=shared</code> or <code>file:memdb1?mode=memory&cache=sha</code></p> <p>Warning: Do not attempt to use the same database Zabbix server is using. In case of Oracle, if the Net Service Name connection method is used, specify the service name from <code>tnsnames.ora</code> or set to empty string; set the <code>TWO_TASK</code> environment variable if DBName is set to empty string.</p>
DBPassword	no			<p>Database password. Ignored for SQLite.</p> <p>Comment this line if no password is used.</p>
DBSchema	no			<p>Schema name. Used for PostgreSQL.</p>
DBSocket	no		3306	<p>Path to MySQL socket.</p> <p>Database port when not using local socket. Ignored for SQLite.</p>
DBUser				<p>Database user. Ignored for SQLite.</p>
DBTLSConnect	no			<p>Setting this option enforces to use TLS connection to database:</p> <ul style="list-style-type: none"> <code>required</code> - connect using TLS <code>verify_ca</code> - connect using TLS and verify certificate <code>verify_full</code> - connect using TLS, verify certificate and verify that database identity specified by DBHost matches its certificate <p>On MySQL starting from 5.7.11 and PostgreSQL the following values are supported: <code>"required"</code>, <code>"verify"</code>, <code>"verify_full"</code>. On MariaDB starting from version 10.2.6 <code>"required"</code> and <code>"verify_full"</code> values are supported.</p> <p>By default not set to any option and the behavior depends on database configuration.</p> <p>This parameter is supported since Zabbix 5.0.0.</p>

Parameter	Mandatory	Range	Default	Description
DBTLSCAFile	no (yes, if DBTLSConnect set to one of: verify_ca, verify_full)			Full pathname of a file containing the top-level CA(s) certificates for database certificate verification. This parameter is supported since Zabbix 5.0.0.
DBTLSCertFile	no			Full pathname of file containing Zabbix server certificate for authenticating to database. This parameter is supported since Zabbix 5.0.0.
DBTLSKeyFile	no			Full pathname of file containing the private key for authenticating to database. This parameter is supported since Zabbix 5.0.0.
DBTLSCipher	no			The list of encryption ciphers that Zabbix server permits for TLS protocols up through TLSv1.2. Supported only for MySQL. This parameter is supported since Zabbix 5.0.0.
DBTLSCipher13	no			The list of encryption ciphersuites that Zabbix server permits for TLSv1.3 protocol. Supported only for MySQL, starting from version 8.0.16. This parameter is supported since Zabbix 5.0.0.
DebugLevel	no	0-5	3	Specifies debug level: 0 - basic information about starting and stopping of Zabbix processes 1 - critical information 2 - error information 3 - warnings 4 - for debugging (produces lots of information) 5 - extended debugging (produces even more information)
EnableRemoteCommands	no		0	Whether remote commands from Zabbix server are allowed. 0 - not allowed 1 - allowed This parameter is supported since Zabbix 3.4.0.
ExternalScripts	no		/usr/local/share/zabbix/external/scripts	Location of external scripts (depends on compile-time installation variable datadir).
Fping6Location	no		/usr/sbin/fping6	Location of fping6. Make sure that fping6 binary has root ownership and SUID flag set. Make empty ("Fping6Location=") if your fping utility is capable to process IPv6 addresses.

Parameter	Mandatory	Range	Default	Description
FpingLocation	no		/usr/sbin/fping	Location of fping. Make sure that fping binary has root ownership and SUID flag set!
HeartbeatFrequency	no	0-3600	60	Frequency of heartbeat messages in seconds. Used for monitoring availability of proxy on server side. 0 - heartbeat messages disabled. Active proxy parameter. Ignored for passive proxies (see ProxyMode parameter).
HistoryCacheSize	no	128K-2G	16M	Size of history cache, in bytes. Shared memory size for storing history data.
HistoryIndexCacheSize	no	128K-2G	4M	Size of history index cache, in bytes. Shared memory size for indexing history data stored in history cache. The index cache size needs roughly 100 bytes to cache one item.
Hostname	no		Set by HostnameItem	This parameter is supported since Zabbix 3.0.0. Unique, case sensitive Proxy name. Make sure the proxy name is known to the server! Allowed characters: alphanumeric, '.', '_', '-' and '-'. Maximum length: 128
HostnameItem	no		system.hostname	Item used for setting Hostname if it is undefined (this will be run on the proxy similarly as on an agent). Does not support UserParameters, performance counters or aliases, but does support system.run[]. Ignored if Hostname is set.

Parameter	Mandatory	Range	Default	Description
HousekeepingFrequency	no	0-24	1	<p>How often Zabbix will perform housekeeping procedure (in hours). Housekeeping is removing outdated information from the database.</p> <p>Note: To prevent housekeeper from being overloaded (for example, when configuration parameters ProxyLocalBuffer or ProxyOfflineBuffer are greatly reduced), no more than 4 times HousekeepingFrequency hours of outdated information are deleted in one housekeeping cycle. Thus, if HousekeepingFrequency is 1, no more than 4 hours of outdated information (starting from the oldest entry) will be deleted per cycle.</p> <p>Note: To lower load on proxy startup housekeeping is postponed for 30 minutes after proxy start. Thus, if HousekeepingFrequency is 1, the very first housekeeping procedure after proxy start will run after 30 minutes, and will repeat every hour thereafter.</p> <p>Since Zabbix 3.0.0 it is possible to disable automatic housekeeping by setting HousekeepingFrequency to 0. In this case the housekeeping procedure can only be started by housekeeper_execute runtime control option and the period of outdated information deleted in one housekeeping cycle is 4 times the period since the last housekeeping cycle, but not less than 4 hours and not greater than 4 days.</p>
Include	no			<p>You may include individual files or all files in a directory in the configuration file.</p> <p>To only include relevant files in the specified directory, the asterisk wildcard character is supported for pattern matching. For example: <code>/absolute/path/to/config/files/*.</code> See special notes about limitations.</p>

Parameter	Mandatory	Range	Default	Description
JavaGateway	no			IP address (or hostname) of Zabbix Java gateway. Only required if Java pollers are started.
JavaGatewayPort	no	1024-32767	10052	Port that Zabbix Java gateway listens on.
ListenBacklog	no	0 - INT_MAX	SOMAXCONN	The maximum number of pending connections in the TCP queue. Default value is a hard-coded constant, which depends on the system. Maximum supported value depends on the system, too high values may be silently truncated to the 'implementation-specified maximum'.
ListenIP	no		0.0.0.0	List of comma delimited IP addresses that the trapper should listen on. Trapper will listen on all network interfaces if this parameter is missing.
ListenPort	no	1024-32767	10051	Listen port for trapper.
LoadModule	no			Module to load at proxy startup. Modules are used to extend functionality of the proxy. Formats: LoadModule=<module.so> LoadModule=<path/module.so> LoadModule=</abs_path/module.so> Either the module must be located in directory specified by LoadModulePath or the path must precede the module name. If the preceding path is absolute (starts with '/') then LoadModulePath is ignored. It is allowed to include multiple LoadModule parameters.
LoadModulePath	no			Full path to location of proxy modules. Default depends on compilation options.
LogFile	yes, if LogType is set to file, otherwise no			Name of log file.
LogFileSize	no	0-1024	1	Maximum size of log file in MB. 0 - disable automatic log rotation. Note: If the log file size limit is reached and file rotation fails, for whatever reason, the existing log file is truncated and started anew.

Parameter	Mandatory	Range	Default	Description
LogRemoteCommands	no		0	Enable logging of executed shell commands as warnings. 0 - disabled 1 - enabled This parameter is supported since Zabbix 3.4.0.
LogType	no		file	Log output type: file - write log to file specified by LogFile parameter, system - write log to syslog, console - write log to standard output. This parameter is supported since Zabbix 3.0.0.
LogSlowQueries	no	0-3600000	0	How long a database query may take before being logged (in milliseconds). 0 - don't log slow queries. This option becomes enabled starting with DebugLevel=3.
PidFile	no		/tmp/zabbix_proxy.pid	Name of PID file.
ProxyLocalBuffer	no	0-720	0	Proxy will keep data locally for N hours, even if the data have already been synced with the server. This parameter may be used if local data will be used by third party applications.
ProxyMode	no	0-1	0	Proxy operating mode. 0 - proxy in the active mode 1 - proxy in the passive mode Note that (sensitive) proxy configuration data may become available to parties having access to the Zabbix server trapper port when using an active proxy. This is possible because anyone may pretend to be an active proxy and request configuration data; authentication does not take place.
ProxyOfflineBuffer	no	1-720	1	Proxy will keep data for N hours in case of no connectivity with Zabbix server. Older data will be lost.
ServerPort	no	1024-32767	10051	Port of Zabbix trapper on Zabbix server. Active proxy parameter. Ignored for passive proxies (see ProxyMode parameter).

Parameter	Mandatory	Range	Default	Description
Server	yes			<p>If ProxyMode is set to active mode: Zabbix server IP address or DNS name (address:port) or cluster (address:port;address2:port) to get configuration data from and send data to. If port is not specified, the default port is used. Cluster nodes must be separated by a semicolon.</p> <p>If ProxyMode is set to passive mode: List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of Zabbix server. Incoming connections will be accepted only from the addresses listed here. If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally. '::/0' will allow any IPv4 or IPv6 address. '0.0.0.0/0' can be used to allow any IPv4 address. Example: Server=127.0.0.1,192.168.1.0/24,::1,2001::</p>
SNMPTrapperFile	no		/tmp/zabbix_traps.tmp	<p>Temporary file used for passing data from SNMP trap daemon to the proxy. Must be the same as in zabbix_trap_receiver.pl or SNMPTT configuration file.</p>
SocketDir	no		/tmp	<p>Directory to store IPC sockets used by internal Zabbix services.</p>
SourceIP	no			<p>This parameter is supported since Zabbix 3.4.0. Source IP address for:</p> <ul style="list-style-type: none"> - outgoing connections to Zabbix server; - agentless connections (VMware, SSH, JMX, SNMP, Telnet and simple checks); - HTTP agent connections; - script item JavaScript HTTP requests; - preprocessing JavaScript HTTP requests; - connections to the Vault
SSHKeyLocation	no			<p>Location of public and private keys for SSH checks and actions</p>
SSLCertLocation	no			<p>Location of SSL client certificate files for client authentication. This parameter is used in web monitoring only.</p>

Parameter	Mandatory	Range	Default	Description
SSLKeyLocation	no			Location of SSL private key files for client authentication. This parameter is used in web monitoring only.
SSLCALocation	no			Location of certificate authority (CA) files for SSL server certificate verification. Note that the value of this parameter will be set as libcurl option CURLOPT_CAPATH. For libcurl versions before 7.42.0, this only has effect if libcurl was compiled to use OpenSSL. For more information see cURL web page . This parameter is used in web monitoring since Zabbix 2.4.0 and in SMTP authentication since Zabbix 3.0.0.
StartDBSyncers	no	1-100	4	Number of pre-forked instances of history syncers . Note: Be careful when changing this value, increasing it may do more harm than good.
StartDiscoverers	no	0-250	1	Number of pre-forked instances of discoverers .
StartHistoryPollers	no	0-1000	1	Number of pre-forked instances of history pollers . This parameter is supported since Zabbix 5.4.0.
StartHTTPPollers	no	0-1000	1	Number of pre-forked instances of HTTP pollers .
StartIPMIPollers	no	0-1000	0	Number of pre-forked instances of IPMI pollers .
StartJavaPollers	no	0-1000	0	Number of pre-forked instances of Java pollers .
StartPingers	no	0-1000	1	Number of pre-forked instances of ICMP pingers .
StartPollersUnreachable	no	0-1000	1	Number of pre-forked instances of pollers for unreachable hosts (including IPMI and Java). At least one poller for unreachable hosts must be running if regular, IPMI or Java pollers are started.
StartPollers	no	0-1000	5	Number of pre-forked instances of pollers .
StartPreprocessors	no	1-1000	3	Number of pre-forked instances of preprocessing workers¹ . The preprocessing manager process is automatically started when a preprocessor worker is started. This parameter is supported since Zabbix 4.2.0.
StartSNMPTrapper	no	0-1	0	If set to 1, SNMP trapper process will be started.

Parameter	Mandatory	Range	Default	Description
StartTrappers	no	0-1000	5	Number of pre-forked instances of trappers . Trappers accept incoming connections from Zabbix sender and active agents.
StartVMwareCollectors	no	0-250	0	Number of pre-forked VMware collector instances.
StatsAllowedIP	no			List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of external Zabbix instances. Stats request will be accepted only from the addresses listed here. If this parameter is not set no stats requests will be accepted. If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally and '::/0' will allow any IPv4 or IPv6 address. '0.0.0.0/0' can be used to allow any IPv4 address. Example: StatsAllowedIP=127.0.0.1,192.168.1.0/24,::1,2001::1 This parameter is supported since Zabbix 4.2.0.
Timeout	no	1-30	3	Specifies how long we wait for agent, SNMP device or external check (in seconds).
TLSAccept	yes for passive proxy, if TLS certificate or PSK parameters are defined (even for unencrypted connection), otherwise no			What incoming connections to accept from Zabbix server. Used for a passive proxy, ignored on an active proxy. Multiple values can be specified, separated by comma: unencrypted - accept connections without encryption (default) psk - accept connections with TLS and a pre-shared key (PSK) cert - accept connections with TLS and a certificate This parameter is supported since Zabbix 3.0.0.
TLSCAFile	no			Full pathname of a file containing the top-level CA(s) certificates for peer certificate verification, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.

Parameter	Mandatory	Range	Default	Description
TLSCertFile	no			Full pathname of a file containing the proxy certificate or certificate chain, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSCipherAll	no			GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for certificate- and PSK-based encryption. Example: TLS_AES_256_GCM_SHA384:TLS_CHACHA20
TLSCipherAll13	no			This parameter is supported since Zabbix 4.4.7. Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for certificate- and PSK-based encryption. Example for GnuTLS: NONE:+VERS- TLS1.2:+ECDHE- RSA:+RSA:+ECDHE- PSK:+PSK:+AES-128- GCM:+AES-128- CBC:+AEAD:+SHA256:+SHA1:+CURVE- ALL:+COMP-NUL::+SIGN- ALL:+CTYPE-X.509 Example for OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128
TLSCipherCert	no			This parameter is supported since Zabbix 4.4.7. GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for certificate-based encryption. Example for GnuTLS: NONE:+VERS- TLS1.2:+ECDHE- RSA:+RSA:+AES-128- GCM:+AES-128- CBC:+AEAD:+SHA256:+SHA1:+CURVE- ALL:+COMP-NUL::+SIGN- ALL:+CTYPE-X.509 Example for OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128

Parameter	Mandatory	Range	Default	Description
TLSCipherCert13	no			Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for certificate-based encryption. This parameter is supported since Zabbix 4.4.7.
TLSCipherPSK	no			GnuTLS priority string or OpenSSL (TLS 1.2) cipher string. Override the default ciphersuite selection criteria for PSK-based encryption. Example for GnuTLS: NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NUL+SIGN-ALL Example for OpenSSL: kECDHEPSK+AES128:kPSK+AES128 This parameter is supported since Zabbix 4.4.7.
TLSCipherPSK13	no			Cipher string for OpenSSL 1.1.1 or newer in TLS 1.3. Override the default ciphersuite selection criteria for PSK-based encryption. Example: TLS_CHACHA20_POLY1305_SHA256:TLS_AES This parameter is supported since Zabbix 4.4.7.
TLSCConnect	yes for active proxy, if TLS certificate or PSK parameters are defined (even for unencrypted connection), otherwise no			How the proxy should connect to Zabbix server. Used for an active proxy, ignored on a passive proxy. Only one value can be specified: unencrypted - connect without encryption (default) psk - connect using TLS and a pre-shared key (PSK) cert - connect using TLS and a certificate This parameter is supported since Zabbix 3.0.0.
TLSCRLFile	no			Full pathname of a file containing revoked certificates. This parameter is used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSSKeyFile	no			Full pathname of a file containing the proxy private key, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.

Parameter	Mandatory	Range	Default	Description
TLSPSKFile	no			Full pathname of a file containing the proxy pre-shared key. used for encrypted communications with Zabbix server. This parameter is supported since Zabbix 3.0.0.
TLSPSKIdentity	no			Pre-shared key identity string, used for encrypted communications with Zabbix server. This parameter is supported since Zabbix 3.0.0.
TLSServerCertIssuer	no			Allowed server certificate issuer. This parameter is supported since Zabbix 3.0.0.
TLSServerCertSubject	no			Allowed server certificate subject. This parameter is supported since Zabbix 3.0.0.
TmpDir	no		/tmp	Temporary directory.
TrapperTimeout	no	1-300	300	Specifies how many seconds trapper may spend processing new data.
User	no		zabbix	Drop privileges to a specific, existing user on the system. Only has effect if run as 'root' and AllowRoot is disabled.
UnavailableDelay	no	1-3600	60	How often host is checked for availability during the unavailability period, in seconds.
UnreachableDelay	no	1-3600	15	How often host is checked for availability during the unreachability period, in seconds.
UnreachablePeriod	no	1-3600	45	After how many seconds of unreachability treat a host as unavailable.
VaultDBPath	no			Vault path from where credentials for database will be retrieved by keys 'password' and 'username'. Example: secret/zabbix/database This option can only be used if DBUser and DBPassword are not specified. This parameter is supported since Zabbix 5.2.0.

Parameter	Mandatory	Range	Default	Description
VaultToken	no			Vault authentication token that should have been generated exclusively for Zabbix proxy with read-only permission to the path specified in the optional VaultDBPath configuration parameter. It is an error if VaultToken and VAULT_TOKEN environment variable are defined at the same time. This parameter is supported since Zabbix 5.2.0.
VaultURL	no		https://127.0.0.1:8200	Vault server HTTP[S] URL. System-wide CA certificates directory will be used if SSLCALocation is not specified. This parameter is supported since Zabbix 5.2.0.
VMwareCacheSize	no	256K-2G	8M	Shared memory size for storing VMware data. A VMware internal check zabbix[vmware,buffer,...] can be used to monitor the VMware cache usage (see Internal checks). Note that shared memory is not allocated if there are no vmware collector instances configured to start.
VMwareFrequency	no	10-86400	60	Delay in seconds between data gathering from a single VMware service. This delay should be set to the least update interval of any VMware monitoring item.
VMwarePerfFrequency	no	10-86400	60	Delay in seconds between performance counter statistics retrieval from a single VMware service. This delay should be set to the least update interval of any VMware monitoring item that uses VMware performance counters.
VMwareTimeout	no	1-300	10	The maximum number of seconds vmware collector will wait for a response from VMware service (vCenter or ESX hypervisor).

3 Agente Zabbix (UNIX)

Visão geral

Esta seção lista os parâmetros suportados em um arquivo de configuração do Zabbix Agent (zabbix_agentd.conf). Note que:

- Os valores padrão refletem os padrões do daemon, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração com codificação UTF-8 sem [BOM](#);

- Comentários começando com “#” são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Alias	não			<p>Configura um apelido para uma chave de item. Ele pode ser usado para substituir uma chave de item longa e complexa com uma menor e mais simples.</p> <p>Múltiplos parâmetros Alias podem estar presente.</p> <p>Múltiplos parâmetros com a mesma chave Alias são permitidos.</p> <p>Diferentes chaves Alias podem referenciar a mesma chave de item.</p> <p>Apelidos podem ser usados nos parâmetros</p> <p>HostMetadataItem mas não em HostNameItem.</p> <p>Exemplos:</p> <p>1. Recuperando o ID do usuário ‘zabbix’.</p> <p>Alias=zabbix.userid:vfs.file.regexp[/etc/passwd/9]+)”,,”\1]</p> <p>Agora a chave abreviada zabbix.userid pode ser usada para recuperar dados.</p> <p>2. Obtendo utilização de CPU com parâmetros padrão e customizados.</p> <p>Alias=cpu.util:system.cpu.util</p> <p>Alias=cpu.util[*]:system.cpu.util[*]</p> <p>Isto permite utilizar a chave cpu.util para obter a porcentagem de utilização de CPU com parâmetros padrão bem como usar cpu.util[all, idle, avg15] para obter dados específicos sobre utilização de CPU.</p> <p>3. Execução de múltiplas regras de descoberta de baixo-nível processando os mesmos itens descobertos.</p> <p>Alias=vfs.fs.discovery[*]:vfs.fs.discovery</p> <p>Agora é possível configurar várias regras de descoberta usando vfs.fs.discovery com diferentes parâmetros para cada regra, p.e., vfs.fs.discovery[foo], vfs.fs.discovery[bar], etc.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
AllowKey	não			<p>Permite a execução daquelas chaves de item que correspondem a um padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" representando qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com DenyKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
AllowRoot	não		0	<p>Permite que o agente seja executado como 'root'. Se desabilitado e o agente for iniciado pelo 'root', o agente tentará trocar para o usuário 'zabbix'. Não tem efeito se iniciado com um usuário regular.</p> <p>0 - não permitir 1 - permitir</p>
BufferSend	não	1-3600	5	<p>Não manter dados por mais do que N segundos no buffer.</p>
BufferSize	não	2-65535	100	<p>Número máximo de valores em um buffer de memória. O agente enviará todos os dados coletados para o Zabbix Server ou Proxy se o buffer estiver cheio.</p>
DebugLevel	não	0-5	3	<p>Especifica o nível de depuração (debug):</p> <p>0 - informações básicas sobre início e parada de processos Zabbix</p> <p>1 - informações críticas</p> <p>2 - informações de erro</p> <p>3 - alertas</p> <p>4 - para depuração (produz bastante informação)</p> <p>5 - depuração estendida (produz ainda mais informação)</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
DenyKey	não			<p>Nega a execução daquelas chaves de item que correspondem a um padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" representando qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com AllowKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
EnableRemoteCommands	não		0	<p>Define se comandos remotos do Zabbix Server são permitidos. Este parâmetro é obsoleto, em vez disso use AllowKey=system.run[*] ou DenyKey=system.run[*]</p> <p>É um apelido interno para os parâmetros AllowKey/DenyKey dependendo do valor: 0 - DenyKey=system.run[*] 1 - AllowKey=system.run[*]</p>
HostInterface	não	0-255 caracteres		<p>Parâmetro opcional que define interface de host. Interface do host é usada no processo de autoregistro. Um agente emitirá um erro e não iniciará se o valor estiver acima do limite de 255 caracteres.</p> <p>Se não definido, o valor será adquirido de HostInterfaceItem.</p> <p>Suportado desde o Zabbix 4.4.0.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
HostInterfaceItem	não			<p>Parâmetro opcional que define um item usado para obtenção de interface de host.</p> <p>Interface do host é usada no processo de autoregistro. Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima de 255 caracteres.</p> <p>Esta opção é usada apenas quando HostInterface não estiver definido.</p> <p>Suportado desde o Zabbix 4.4.0.</p>
HostMetadata	não	0-255 caracteres		<p>Parâmetro opcional que define metadado do host. O metadado do host é usado apenas no processo de autoregistro (agente ativo). Se não definido, o valor será obtido de HostMetadataItem. Um agente emitirá um erro e não iniciará se o valor especificado estiver acima do limite ou uma string não-UTF-8.</p>
HostMetadataItem	não			<p>Parâmetro opcional que define a item Zabbix Agent usado para obtenção de metadado de host. Esta opção é usada apenas quando HostMetadata não é definido. Suporta UserParameters e apelidos (aliases). Suporta system.run[] independente dos valores de AllowKey/DenyKey.</p> <p>O valor de HostMetadataItem é recuperado em cada tentativa de autoregistro e é usado somente no processo de autoregistro do host (agente ativo).</p> <p>Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima do limite de 255 caracteres.</p> <p>O valor retornado pelo item deve ser uma string UTF-8 caso contrário ele será ignorado.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Hostname	não		Configurado por HostnameItem	<p>Lista separada por vírgula de nomes de host únicos, sensíveis a maiúsculas e minúsculas.</p> <p>Necessário para verificações ativas e deve corresponder aos nomes de host conforme configurados no Server. O valor é adquirido de HostnameItem se não definido.</p> <p>Caracteres permitidos: alfanumérico, '.', ',', '_' e '-'. Comprimento máximo: 128 caracteres por nome de host, 2048 caracteres para a linha inteira.</p>
HostnameItem	não		system.hostname	<p>Parâmetro opcional que define um item Zabbix Agent usado para obtenção de nome de host. Esta opção é usada apenas quando Hostname não for definido. Não suporta UserParameters ou apelidos (aliases), mas suporta system.run[] independente dos valores de AllowKey/DenyKey. O comprimento de saída é limitado em 512KB.</p>
Include	não			<p>Você pode incluir arquivos individuais ou todos os arquivos de um diretório no arquivo de configuração. Para incluir apenas arquivos relevantes no diretório especificado, o caracter curinga asterisco é suportado para correspondência de padrão. Por exemplo: /absolute/path/to/config/files/*. Consulte notas especiais sobre limitações.</p>
ListenBacklog	não	0 - INT_MAX	SOMAXCONN	<p>O número máximo de conexões pendentes na fila TCP.</p> <p>O valor padrão é uma constante gravada em código, que depende do sistema.</p> <p>O valor máximo suportado depende do sistema, valores muito altos podem ser silenciosamente truncados para o 'máximo especificado na implementação'.</p>
ListenIP	não		0.0.0.0	<p>Lista separada por vírgula de endereços IP que o agente deve ouvir.</p> <p>Múltiplos endereços IP são suportados na versão 1.8.3 e superior.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
ListenPort	não	1024-32767	10050	O Agent ouvirá nesta porta por conexões do Server.
LoadModule	não			Módulo para carregar na inicialização do agente. Módulos são usados para estender a funcionalidade do agente. Formatos: LoadModule=<module.so> LoadModule=<path/module.so> LoadModule=</abs_path/module.so> Ou o módulo deve estar localizado no diretório especificado por LoadModulePath ou o caminho deve preceder o nome do módulo. Se o caminho precedente for absoluto (começa com '/') então LoadModulePath é ignorado. É permitido incluir múltiplos parâmetros LoadModule.
LoadModulePath	não			Caminho completo para a localização dos módulos do agente. O padrão depende das opções de compilação.
LogFile	sim, se LogType estiver configurado para file (arquivo), caso contrário não			Nome do arquivo de log.
LogFileSize	não	0-1024	1	Tamanho máximo do arquivo de log em MB. 0 - desabilita rotação automática de log. Nota: Se o limite de tamanho de arquivo de log for alcançado e a rotação de arquivo falhar, por qualquer razão, o arquivo existente é truncado e iniciado outra vez.
LogType	não		file	Tipo de saída de log: file - grava o log em um arquivo especificado pelo parâmetro LogFile, system - grava o log no syslog, console - grava o log na saída padrão. Este parâmetro é suportado desde o Zabbix 3.0.0.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
LogRemoteCommands	não		0	Habilita o registro em log dos comandos shell executados como alertas. 0 - desabilitado 1 - habilitado Comandos serão registrados em log apenas se executados remotamente. Entradas de log não serão criadas se system.run[] for lançado localmente pelo parâmetro HostMetadataltem, HostInterfaceltem ou Hostnameltem parameters.
MaxLinesPerSecond	não	1-1000	20	Número máximo de novas linhas que o agente enviará por segundo para o Zabbix Server ou Proxy quando processando verificações ativas de 'log' e 'eventlog'. O valor fornecido será sobrescrito pelo parâmetro 'maxlines', fornecido na chave de item 'log' ou 'eventlog'. Nota: O Zabbix processará 10 vezes mais novas linhas do que configurado em MaxLinesPerSecond para procurar pela string requisitada nos itens de log.
PidFile	não		/tmp/zabbix_agentd.pid	Nome do arquivo PID.
RefreshActiveChecks	não	60-3600	120	Com que frequência a lista de verificações ativas é atualizada, em segundos. Note que após falhar em atualizar as verificações ativas a próxima atualização será tentada após 60 segundos.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Server	sim, se StartAgents não estiver configurado explicitamente como 0			<p>Lista separada por vírgula de endereços IP, opcionalmente em notação CIDR, ou nomes de host de Zabbix Servers e Zabbix Proxies.</p> <p>Conexões de entrada serão aceitas apenas dos host listados aqui.</p> <p>Se o suporte a IPv6 estiver habilitado então '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' são tratados igualmente e '::/0' permitirá qualquer endereço IPv4 ou IPv6. '0.0.0.0/0' pode ser usado para permitir qualquer endereço IPv4.</p> <p>Note que "endereços IPv6 compatíveis com IPv4" (0000::/96 prefix) são suportados mas obsoletos pela RFC4291.</p> <p>Exemplo: Server=127.0.0.1,192.168.1.0/24,::1,2001: Espaços são permitidos.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
ServerActive	não			<p>Lista separada por vírgula de pares de endereços IP ou nomes DNS (endereço:porta) ou clusters (endereço:porta;endereço2:porta) de Zabbix Servers e Zabbix Proxies para verificações ativas.</p> <p>Nós de cluster devem ser separados por um ponto-e-vírgula.</p> <p>Múltiplos endereços separados por vírgula podem ser fornecidos para usar vários Zabbix Servers independentes em paralelo. Espaços são permitidos.</p> <p>Se uma porta não for especificada, a porta padrão é usada.</p> <p>Endereços IPv6 devem ser envoltos por colchetes se a porta para o host for especificada.</p> <p>Se porta não for especificada, colchetes para endereços IPv6 são opcionais.</p> <p>Se este parâmetro não for especificado, verificações ativas estão desabilitadas.</p> <p>Exemplo para múltiplos Servers:</p> <p>ServerActive=127.0.0.1:20051,zabbix.doma</p> <p>Exemplo para alta disponibilidade:</p> <p>ServerActive=zabbix.cluster.node1;zabbix.c</p> <p>Exemplo para alta disponibilidade com dois clusters e um Server:</p> <p>ServerActive=zabbix.cluster.node1;zabbix.c</p>
SourceIP	não			<p>Endereço IP de origem para:</p> <ul style="list-style-type: none"> - conexões de saída para Zabbix Server ou Zabbix Proxy; - criação de conexões enquanto executando alguns itens (web.page.get, net.tcp.port, etc.)
StartAgents	não	0-100	3	<p>Número de instâncias separadas (pre-forked) do zabbix_agentd que processa verificações passivas.</p> <p>Se configurado como 0, desabilita verificações passivas e o agente não ouvirá em nenhuma porta TCP.</p>
Timeout	não	1-30	3	<p>Gaste não mais do que Timeout segundos no processamento.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSAccept	sim, se certificado TLS ou parâmetros PSK são definidos (mesmo para conexão não criptografada), caso contrário não			Quais conexões de entrada aceitar. Usado para verificações passivas. Múltiplos valores podem ser especificados, separados por vírgula: unencrypted - aceita conexões sem criptografia (padrão) psk - aceita conexões com TLS e chave pré-compartilhada (PSK) cert - aceita conexões com TLS e um certificado Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSCAFile	não			Caminho completo de um arquivo contendo os certificados CA(s) de topo-de-nível para verificação de pares de certificado, usado para comunicações criptografadas entre componentes Zabbix. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSCertFile	não			Caminho completo de um arquivo contendo o certificado de agente ou cadeia de certificado, usado para comunicações criptografadas com componentes Zabbix. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSCipherAll	não			String de prioridade GnuTLS ou string de cifra OpenSSL (TLS 1.2). Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em certificado e PSK. Exemplo: TLS_AES_256_GCM_SHA384:TLS_CHACHA20 Este parâmetro é suportado desde o Zabbix 4.4.7.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSCipherAll13	não			String de cifra para OpenSSL 1.1.1 ou mais recente em TLS 1.3. Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em certificado e PSK. Exemplo para GnuTLS: NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL::+SIGN-ALL:+CTYPE-X.509 Exemplo para OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128 Este parâmetro é suportado desde o Zabbix 4.4.7.
TLSCipherCert	não			String de prioridade GnuTLS ou string de cifra OpenSSL (TLS 1.2). Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em certificado. Exemplo para GnuTLS: NONE:+VERS-TLS1.2:+ECDHE-RSA:+RSA:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NULL:+SIGN-ALL:+CTYPE-X.509 Exemplo para OpenSSL: EECDH+aRSA+AES128:RSA+aRSA+AES128 Este parâmetro é suportado desde o Zabbix 4.4.7.
TLSCipherCert13	não			String de cifra para OpenSSL 1.1.1 ou mais recente em TLS 1.3. Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em certificado. Este parâmetro é suportado desde o Zabbix 4.4.7.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSCipherPSK	não			String de prioridade GnuTLS ou string de cifra OpenSSL (TLS 1.2). Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em PSK. Exemplo para GnuTLS: NONE:+VERS-TLS1.2:+ECDHE-PSK:+PSK:+AES-128-GCM:+AES-128-CBC:+AEAD:+SHA256:+SHA1:+CURVE-ALL:+COMP-NUL:+SIGN-ALL Exemplo para OpenSSL: kECDHEPSK+AES128:kPSK+AES128 Este parâmetro é suportado desde o Zabbix 4.4.7.
TLSCipherPSK13	não			String de cifra para OpenSSL 1.1.1 ou mais recente em TLS 1.3. Sobrescreve o critério de seleção de suíte de cifra padrão para criptografia baseada em PSK. Exemplo: TLS_CHACHA20_POLY1305_SHA256:TLS_AES Este parâmetro é suportado desde o Zabbix 4.4.7.
TLSConnect	sim, se certificado TLS ou parâmetros PSK estão definidos (mesmo para conexão não criptografada), caso contrário não			Como o agente deve conectar ao Zabbix Server ou Proxy. Usado para verificações ativas. Apenas um valor pode ser especificado: unencrypted - conecta sem criptografia (padrão) psk - conecta usando TLS e uma chave pré-compartilhada (PSK) cert - conecta usando TLS e um certificado Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSCRLFile	não			Caminho completo de um arquivo contendo certificados revogados. Este parâmetro é usado para comunicações criptografadas com componentes Zabbix. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSKeyFile	não			Caminho completo de um arquivo contendo a chave privada do agente usada para comunicação criptografada com componentes Zabbix. Este parâmetro é suportado desde o Zabbix 3.0.0.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSPSKFile	não			Caminho completo de um arquivo contendo a chave pré-compartilhada do agente usada para comunicação criptografada com componentes Zabbix. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSPSKIdentity	não			String de identidade de chave pré-compartilhada, usada para comunicações criptografadas com Zabbix Server. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSServerCertIssuer	não			Emissor de certificado de Server (proxy) permitido. Este parâmetro é suportado desde o Zabbix 3.0.0.
TLSServerCertSubject	não			Cliente de certificado de Server (proxy) permitido. Este parâmetro é suportado desde o Zabbix 3.0.0.
UnsafeUserParameters	não	0,1	0	Permite que todos os caracteres sejam passados em argumentos para parâmetros definidos por usuário. 0 - não permitir 1 - permitir Os seguintes caracteres não são permitidos: \\ ' " * ? [] { } ~ \$! & ; () > # @ Adicionalmente, caracteres de nova linha não são permitidos.
User	não		zabbix	Revoga privilégios para um usuário específico, existente no sistema. Tem efeito apenas se executado como 'root' e AllowRoot estiver desabilitado.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
UserParameter	não			<p>Parâmetro definido por usuário para monitoramento. Podem existir vários parâmetros definidos por usuário.</p> <p>Formato: UserParameter=<chave>,<comando shell></p> <p>Note que o comando shell não deve retornar um string vazia ou apenas EOL.</p> <p>Comandos shell podem ter caminhos relativos, se o parâmetro UserParameterDir for especificado.</p> <p>Exemplos:</p> <p>UserParameter=system.test,who wc -l</p> <p>UserParameter=check_cpu,./custom_script.</p>
UserParameterDir	não			<p>Caminho de busca padrão para comandos UserParameter. Se usado, o agente alterará seu diretório de operação para o especificado aqui antes de executar um comando. Deste modo, comandos UserParameter podem ter um prefixo relativo ./ em vez de uma caminho completo.</p> <p>Apenas uma entrada é permitida.</p> <p>Exemplo: UserParameterDir=/opt/myscripts</p>

Veja também

1. [Diferenças na configuração do Zabbix Agent para verificações ativas e passivas a partir da versão 2.0.0](#)

4 Zabbix Agent 2 (UNIX)

Visão geral

O Zabbix Agent 2 é uma nova geração de agente Zabbix e pode ser usado no lugar do Zabbix Agent.

Esta seção lista os parâmetros suportados em um arquivo de configuração do Zabbix Agent 2 (zabbix_agent2.conf).

Observe que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados somente no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Alias	não			<p>Configura um apelido para uma chave de item. Ele pode ser usado para substituir uma chave de item longa e complexa por uma menor e mais simples. Múltiplos parâmetros Alias podem estar presentes. Múltiplos parâmetros com a mesma chave Alias são permitidos. Diferentes chaves Alias podem referenciar a mesma chave de item. Apelidos podem ser usados nos parâmetros HostMetadataItem mas não nos HostnameItem.</p> <p>Exemplos:</p> <p>1. Recuperando o ID do usuário 'zabbix'. Alias=zabbix.userid:vfs.file.regexp[/etc/passwd/9]+)"",\1] Agora a chave abreviada zabbix.userid pode ser usada para recuperação deste dado.</p> <p>2. Obtendo utilização de CPU com parâmetros padrão e customizados. Alias=cpu.util:system.cpu.util Alias=cpu.util[*]:system.cpu.util[*] Isto permite usar a chave cpu.util para obter a porcentagem de utilização de CPU com parâmetros padrão assim como usar cpu.util[all, idle, avg15] para obter dados específicos sobre utilização de CPU.</p> <p>3. Execução de múltiplas regras de descoberta de baixo-nível processando os mesmos itens descobertos. Alias=vfs.fs.discovery[*]:vfs.fs.discovery Agora é possível configurar várias regras de descoberta usando vfs.fs.discovery com parâmetros diferentes para cada regra, p.e., vfs.fs.discovery[foo], vfs.fs.discovery[bar], etc.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
AllowKey	não			<p>Permite execução das chaves de item que correspondem ao padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" para corresponder a qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com DenyKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
BufferSend	não	1-3600	5	<p>O intervalo de tempo em segundos que determina com que frequência os valores são enviados do buffer para o Zabbix Server. Note que se o buffer estiver cheio, os dados serão enviados mais cedo.</p>
BufferSize	não	2-65535	100	<p>Número máximo de valores em um buffer de memória. O agente enviará todos os dados coletados para o Zabbix Server ou Proxy se o buffer estiver cheio.</p> <p>Este parâmetro deve ser usado apenas se o buffer persistente estiver desabilitado (EnablePersistentBuffer=0).</p>
ControlSocket	não		/tmp/agent.sock	<p>O socket de controle, usado para enviar comandos de tempo de execução com a opção '-R'.</p>
DebugLevel	não	0-5	3	<p>Especifica nível de depuração (debug):</p> <ul style="list-style-type: none"> 0 - informações básicas sobre início e parada dos processos do Zabbix 1 - informações críticas 2 - informações de erro 3 - alertas 4 - para depuração (produz bastante informação) 5 - depuração estendida (produz ainda mais informações)

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
DenyKey	não			<p>Nega a execução das chaves de item que correspondem ao padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" para corresponder a qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com DenyKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
EnablePersistentBuffer	não	0-1	0	<p>Habilita o uso de armazenamento local persistente para itens ativos.</p> <p>0 - desabilitado 1 - habilitado</p> <p>Se o armazenamento persistente estiver desabilitado, o buffer de memória será usado.</p>
HostInterface	não	0-255 caracteres		<p>Parâmetro opcional que define interface do host. A interface do host é usada no processo de autoregistro de host.</p> <p>Um agente emitirá um erro e não iniciará se o valor estiver acima do limite de 255 caracteres.</p> <p>Se não definido, o valor será adquirido de HostInterfaceltem.</p> <p>Suportado desde o Zabbix 4.4.0.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
HostInterfaceItem	não			<p>Parâmetro opcional que define um item usado para obtenção da interface do host.</p> <p>A interface do host é usada no processo de autoregistro de host.</p> <p>Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima do limite de 255 caracteres. Esta opção é usada apenas quando HostInterface não é definido.</p> <p>Suportado desde o Zabbix 4.4.0.</p>
HostMetadata	não	0-255 caracteres		<p>Parâmetro opcional que define os metadados do host. Os metadados de host são usados no processo de autoregistro.</p> <p>Um agente emitirá um erro e não iniciará se o valor especificado estiver acima do limite ou for uma string não-UTF-8.</p> <p>Se não definido, o valor será adquirido de HostMetadataItem.</p>
HostMetadataItem	não			<p>Parâmetro opcional que define um item usado para obtenção de metadados de host. O item de metadados de host é recuperado em cada tentativa de autoregistro para o processo de autoregistro de host.</p> <p>Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima do limite de 255 caracteres. Esta opção é usada apenas quando HostMetadata não é definido.</p> <p>Suporta UserParameters e apelidos (aliases). Suporta system.run[] independentemente dos valores de AllowKey/DenyKey.</p> <p>O valor retornado pelo item deve ser uma string UTF-8 caso contrário ele será ignorado.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Hostname	não		Configurado por HostnameItem	<p>Lista separada por vírgula de nomes de host únicos, sensíveis a maiúsculas e minúsculas.</p> <p>Necessário para verificações ativas e deve corresponder aos nomes de host conforme configurado no Server. O valor é adquirido de HostnameItem se não definido.</p> <p>Caracteres permitidos: alfanuméricos, '.', ',', '_', e '-'.</p> <p>Comprimento máximo: 128 caracteres por nome de host, 2048 caracteres para a linha inteira.</p>
HostnameItem	não		system.hostname	<p>Item usado para geração do Hostname se ele não estiver definido. Ignorado se Hostname estiver definido.</p> <p>Não suporta UserParameters ou apelidos, mas suporta system.run[] independentemente dos valores de AllowKey/DenyKey.</p> <p>O comprimento de saída é limitado em 512KB.</p>
Include	não			<p>Você pode incluir arquivos individuais ou todos os arquivos de um diretório no arquivo de configuração.</p> <p>Durante a instalação o Zabbix criará o diretório de include em /usr/local/etc, a menos que modificado durante o tempo de compilação.</p> <p>Para incluir apenas arquivos relevantes no diretório especificado, o caracter curinga asterisco é suportado para correspondência de padrão.</p> <p>Por exemplo: /absolute/path/to/config/files/*</p> <p>Desde o Zabbix 6.0.0 um caminho pode ser relativo à localização do arquivo zabbix_agent2.conf.</p> <p>Consulte notas especiais sobre limitações.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
ListenIP	não		0.0.0.0	Lista separada por vírgula de endereços IP que o agente deve ouvir. O primeiro endereço IP é enviado para o Zabbix Server, se conectando a ele, para recuperar a lista de verificações ativas.
ListenPort	não	1024-32767	10050	O agente ouvirá nesta porta por conexões vindas do Server.
LogFile	sim, se LogType estiver configurado como file, caso contrário não		/tmp/zabbix_agent2.log	Nome do arquivo de log se LogType for 'file'.
LogFileSize	não	0-1024	1	Tamanho máximo do arquivo de log em MB. 0 - desabilita rotação de log automática. Nota: Se o limite de tamanho de arquivo de log for alcançado e a rotação do arquivo falhar, por qualquer razão, o arquivo de log existente é truncado e iniciado outra vez.
LogType	não		file	Especifica onde as mensagens de log são gravadas: system - syslog, file - arquivo especificado pelo parâmetro LogFile, console - saída padrão.
PersistentBufferFile	não			O arquivo, onde o Zabbix Agent2 deve manter o banco de dados SQLite. Deve ser um nome de arquivo completo. Este parâmetro é usado apenas se o buffer persistente estiver habilitado (EnablePersistentBuffer=1).
PersistentBufferPeriod	não	1m-365d	1h	O período de tempo pelo qual o dado deve ser armazenado, quando não há conexão com o Server ou Proxy. Dados mais antigos serão perdidos. Dados de log serão preservados. Este parâmetro é usado somente se o buffer persistente estiver habilitado (EnablePersistentBuffer=1).
PidFile	não		/tmp/zabbix_agent2.pid	Nome do arquivo PID.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins	não			Desde o Zabbix 6.0.0 a maioria dos plugins tem seus próprios arquivos de configuração . O arquivo de configuração contém os parâmetros de plugin listados abaixo.
	Plugins.log.MaxLinesPerSecond	1-1000	20	Número máximo de novas linhas que o agente enviará por segundo para o Zabbix Server ou Proxy quando processando verificações ativas de 'log' e 'eventlog'. O valor fornecido será sobrescrito pelo parâmetro 'maxlines', provido na chave de item 'log' ou 'eventlog'. Nota: O Zabbix processará 10 vezes mais novas linhas que o configurado em MaxLinesPerSecond para procurar a string requerida em itens de log. Este parâmetro é suportado desde o 4.4.2 e substitui MaxLinesPerSecond.
	Plugins.systemRun.LogRemoteCommands		0	Habilita registro de log de comandos shell executados como alertas. 0 - desabilitado 1 - habilitado Comandos serão registrados em log apenas se executados remotamente. Entradas de log não serão criadas se system.run[] for lançado localmente pelos parâmetros HostMetadataItem, HostInterfaceItem ou HostnameItem. Este parâmetro é suportado desde o 4.4.2 e substitui LogRemoteCommands.
RefreshActiveChecks	não	60-3600	120	Com que frequência a lista de verificações ativas é atualizada, em segundos. Note que após uma falha em atualizar as verificações ativas a próxima atualização será tentada após 60 segundos.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Server	sim			<p>Lista separada por vírgula de endereços IP, opcionalmente em notação CIDR, ou nomes DNS de Zabbix Servers e Zabbix Proxies.</p> <p>Conexões de entrada serão aceitas apenas a partir dos hosts listados aqui.</p> <p>Se o suporte a IPv6 estiver habilitado então '127.0.0.1', '::ffff:127.0.0.1' são tratados igualmente e '::/0' permitirão qualquer endereço IPv4 ou IPv6. '0.0.0.0/0' pode ser usado para permitir quaisquer endereços IPv4.</p> <p>Exemplo: Server=127.0.0.1,192.168.1.0/24,::1,2001</p>
ServerActive	não			<p>Espaços são permitidos.</p> <p>Lista separada por vírgula de pares de endereços IP ou nome DNS (endereço:porta) ou clusters (endereço:porta;endereço2:porta) de Zabbix Servers e Zabbix Proxies para verificações ativas.</p> <p>Nós de cluster devem ser separados por um ponto-e-vírgula.</p> <p>Múltiplos endereços separados por vírgula podem ser fornecidos para usar vários Zabbix Servers independentes em paralelo.</p> <p>Espaços são permitidos.</p> <p>Se porta não for especificada, a porta padrão é usada.</p> <p>Endereços IPv6 devem ser envoltos por colchetes se a porta para o host for especificado.</p> <p>Se a porta não for especificada, os colchetes para endereços IPv6 são opcionais.</p> <p>Se este parâmetro não for especificado, verificações ativas são desabilitadas.</p> <p>Exemplo para múltiplos Servers: ServerActive=127.0.0.1:20051,zabbix.don</p> <p>Exemplo para alta disponibilidade: ServerActive=zabbix.cluster.node1;zabbix</p> <p>Exemplo para alta disponibilidade com dois clusters e um Server: ServerActive=zabbix.cluster.node1;zabbix</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
SourceIP	não			Endereços IP de origem para: - conexões de saída para o Zabbix Server ou Zabbix Proxy; - criação de conexões quando executando alguns itens (web.page.get, net.tcp.port, etc.)
StatusPort	não	1024-32767		Se configurado, o agente ouvirá nesta porta por requisições de estado HTTP (http://localhost:<porta>/status).
Timeout	não	1-30	3	Gastar não mais do que Timeout segundos no processamento.
TLSAccept	sim, se certificado TLS ou parâmetros PSK são definidos (mesmo para conexão não criptografada), caso contrário não			Quais conexões de entrada aceitar. Usado para verificações passivas. Múltiplos valores podem ser especificados, separados por vírgula: unencrypted - aceita conexões sem criptografia (padrão) psk - aceita conexões com TLS e uma chave pré-compartilhada (PSK) cert - aceita conexões com TLS e um certificado
TLSCAFile	não			Caminho completo de um arquivo contendo certificados CA(s) de topo-de-nível para verificação de pares de certificado, usado para comunicações criptografadas entre componentes Zabbix.
TLSCertFile	não			Caminho completo de um arquivo contendo os certificados de agente ou cadeia de certificado, usado para comunicações criptografadas com componentes Zabbix.
TLSConnect	sim, se certificado TLS ou parâmetros PSK estão definidos (mesmo para conexões não criptografadas), caso contrário não			Como o agente deve se conectar ao Zabbix Server ou Proxy. Usado para verificações ativas. Apenas um valor pode ser especificado: unencrypted - conecta sem criptografia (padrão) psk - conecta usando TLS e uma chave pré-compartilhada (PSK) cert - conecta usando TLS e um certificado

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSCRLFile	não			Caminho completo de um arquivo contendo certificados revogados. Este parâmetro é usado para comunicações criptografadas com componentes Zabbix.
TLSKeyFile	não			Caminho completo de um arquivo contendo a chave privada do agente usada para comunicações criptografadas com componentes Zabbix.
TLSPSKFile	não			Caminho completo de um arquivo contendo a chave pré-compartilhada do agente usada para comunicações criptografadas com componentes Zabbix.
TLSPSKIdentity	não			String de identidade de chave pré-compartilhada, usada para comunicações criptografadas com o Zabbix Server.
TLSSEServerCertIssuer	não			Emissor de certificado de Server (Proxy) permitido.
TLSSEServerCertSubject	não			Cliente de certificado de Server (Proxy) permitido.
UnsafeUserParameters	não	0,1	0	<p>Permite que todos os caracteres sejam passados em argumentos para parâmetros definidos por usuário.</p> <p>Os seguintes caracteres não são permitidos:</p> <pre>\ ' " * ? [] { } ~ \$! & ; () > # @</pre> <p>Adicionalmente, caracteres de nova linha não são permitidos.</p>
UserParameter	não			<p>Parâmetro definido por usuário para monitoramento. Podem existir vários parâmetros definidos por usuário.</p> <p>Formato: UserParameter=<chave>,<comando shell></p> <p>Note que o comando shell não deve retornar uma string vazia ou apenas EOL. Comandos shell podem ter caminhos relativos, se o parâmetro UserParameterDir for especificado.</p> <p>Exemplos:</p> <pre>UserParameter=system.test,who wc -l</pre> <pre>UserParameter=check_cpu,./custom_script</pre>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
UserParameterDir	não			<p>Caminho de busca padrão para comandos UserParameter. Se usado, o agente alterará seu diretório de operação para o especificado aqui antes da execução de um comando. Deste modo, comandos UserParameter podem ter um prefixo relativo ./ em vez de um caminho completo.</p> <p>Apenas uma entrada é permitida.</p> <p>Exemplo: UserParameter-Dir=/opt/myscripts</p>

5 Zabbix agent (Windows)

Visão geral

Esta seção lista os parâmetros suportados em um arquivo de configuração do agente Zabbix (Windows) (zabbix_agent.conf).

Observe que:

- Os valores padrão refletem os padrões do daemon, não os valores nos arquivos de configuração enviados;
- O Zabbix suporta arquivos de configuração apenas em codificação UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parameters

Parameter	Mandatory	Range	Default	Description
Alias	no			<p>Sets an alias for an item key. It can be used to substitute long and complex item key with a smaller and simpler one.</p> <p>Multiple Alias parameters may be present. Multiple parameters with the same Alias key are allowed. Different Alias keys may reference the same item key. Aliases can be used in HostMetadataItem but not in HostNameItem or PerfCounter parameters.</p> <p>Examples:</p> <ol style="list-style-type: none"> Retrieving paging file usage in percents from the server. Alias=pg_usage:perf_counter[\\Paging File(_Total)\\% Usage] Now shorthand key pg_usage may be used to retrieve data. Getting CPU load with default and custom parameters. Alias=cpu.load:system.cpu.load Alias=cpu.load[*]:system.cpu.load[*] This allows use cpu.load key to get CPU utilization percentage with default parameters as well as use cpu.load[percpu,avg15] to get specific data about CPU load. Running multiple low-level discovery rules processing the same discovery items. Alias=vfs.fs.discovery[*]:vfs.fs.discovery Now it is possible to set up several discovery rules using vfs.fs.discovery with different parameters for each rule, e.g., vfs.fs.discovery[foo], vfs.fs.discovery[bar], etc.

Parameter	Mandatory	Range	Default	Description
AllowKey	no			<p>Allow execution of those item keys that match a pattern. Key pattern is a wildcard expression that supports "*" character to match any number of any characters. Multiple key matching rules may be defined in combination with DenyKey. The parameters are processed one by one according to their appearance order. This parameter is supported since Zabbix 5.0.0. See also: Restricting agent checks.</p>
BufferSend	no	1-3600	5	Do not keep data longer than N seconds in buffer.
BufferSize	no	2-65535	100	Maximum number of values in a memory buffer. The agent will send all collected data to Zabbix server or proxy if the buffer is full.
DebugLevel	no	0-5	3	<p>Specifies debug level:</p> <ul style="list-style-type: none"> 0 - basic information about starting and stopping of Zabbix processes 1 - critical information 2 - error information 3 - warnings 4 - for debugging (produces lots of information) 5 - extended debugging (produces even more information)
DenyKey	no			<p>Deny execution of those item keys that match a pattern. Key pattern is a wildcard expression that supports "*" character to match any number of any characters. Multiple key matching rules may be defined in combination with AllowKey. The parameters are processed one by one according to their appearance order. This parameter is supported since Zabbix 5.0.0. See also: Restricting agent checks.</p>

Parameter	Mandatory	Range	Default	Description
EnableRemoteCommands	no		0	Whether remote commands from Zabbix server are allowed. This parameter is deprecated , use AllowKey=system.run[*] or DenyKey=system.run[*] instead It is internal alias for AllowKey/DenyKey parameters depending on value: 0 - DenyKey=system.run[*] 1 - AllowKey=system.run[*].
HostInterface	no	0-255 characters		Optional parameter that defines host interface. Host interface is used at host autoregistration process. An agent will issue an error and not start if the value is over the limit of 255 characters. If not defined, value will be acquired from HostInterfaceItem.
HostInterfaceItem	no			Supported since Zabbix 4.4.0. Optional parameter that defines an item used for getting host interface. Host interface is used at host autoregistration process. During an autoregistration request an agent will log a warning message if the value returned by specified item is over limit of 255 characters. This option is only used when HostInterface is not defined.
HostMetadata	no	0-255 characters		Supported since Zabbix 4.4.0. Optional parameter that defines host metadata. Host metadata is used only at host autoregistration process (active agent). If not defined, the value will be acquired from HostMetadataItem. An agent will issue an error and not start if the specified value is over the limit or a non-UTF-8 string.

Parameter	Mandatory	Range	Default	Description
HostMetadataItem	no			<p>Optional parameter that defines a Zabbix agent item used for getting host metadata. This option is only used when HostMetadata is not defined.</p> <p>Supports UserParameters, performance counters and aliases. Supports system.run[] regardless of EnableRemoteCommands value.</p> <p>HostMetadataItem value is retrieved on each autoregistration attempt and is used only at host autoregistration process (active agent).</p> <p>During an autoregistration request an agent will log a warning message if the value returned by the specified item is over the limit of 255 characters.</p> <p>The value returned by the item must be a UTF-8 string otherwise it will be ignored.</p>
Hostname	no		Set by HostnameItem	<p>List of comma-delimited unique, case-sensitive hostnames.</p> <p>Required for active checks and must match hostnames as configured on the server. Value is acquired from HostnameItem if undefined.</p> <p>Allowed characters: alphanumeric, '.', '-', '_' and '-'. Maximum length: 128 characters per hostname, 2048 characters for the entire line.</p>
HostnameItem	no		system.hostname	<p>Optional parameter that defines a Zabbix agent item used for getting host name. This option is only used when Hostname is not defined.</p> <p>Does not support UserParameters, performance counters or aliases, but does support system.run[] regardless of EnableRemoteCommands value.</p> <p>The output length is limited to 512KB.</p> <p>See also a more detailed description.</p>

Parameter	Mandatory	Range	Default	Description
Include	no			<p>You may include individual files or all files in a directory in the configuration file.</p> <p>To only include relevant files in the specified directory, the asterisk wildcard character is supported for pattern matching. For example: <code>/absolute/path/to/config/files/*.</code></p> <p>See special notes about limitations.</p>
ListenBacklog	no	0 - INT_MAX	SOMAXCONN	<p>The maximum number of pending connections in the TCP queue.</p> <p>Default value is a hard-coded constant, which depends on the system.</p> <p>Maximum supported value depends on the system, too high values may be silently truncated to the 'implementation-specified maximum'.</p>
ListenIP	no		0.0.0.0	<p>List of comma-delimited IP addresses that the agent should listen on.</p>
ListenPort	no	1024-32767	10050	<p>Agent will listen on this port for connections from the server.</p>
LogFile	yes, if LogType is set to file, otherwise no		C:\zabbix_agentd.log	<p>Name of the agent log file.</p>
LogFileSize	no	0-1024	1	<p>Maximum size of log file in MB.</p> <p>0 - disable automatic log rotation.</p> <p>Note: If the log file size limit is reached and file rotation fails, for whatever reason, the existing log file is truncated and started anew.</p>
LogType	no		file	<p>Log output type:</p> <p>file - write log to file specified by LogFile parameter, system - write log Windows Event Log, console - write log to standard output.</p> <p>This parameter is supported since Zabbix 3.0.0.</p>
LogRemoteCommands	no		0	<p>Enable logging of executed shell commands as warnings.</p> <p>0 - disabled 1 - enabled</p>

Parameter	Mandatory	Range	Default	Description
MaxLinesPerSecond	no	1-1000	20	<p>Maximum number of new lines the agent will send per second to Zabbix server or proxy processing 'log', 'logrt' and 'eventlog' active checks.</p> <p>The provided value will be overridden by the parameter 'maxlines', provided in 'log', 'logrt' or 'eventlog' item keys.</p> <p>Note: Zabbix will process 10 times more new lines than set in MaxLinesPerSecond to seek the required string in log items.</p>
PerfCounter	no			<p>Defines a new parameter <parameter_name> which is an average value for system performance counter <perf_counter_path> for the specified time period <period> (in seconds).</p> <p>Syntax: <parameter_name>,"<perf_counter_path>",<period></p> <p>For example, if you wish to receive average number of processor interrupts per second for last minute, you can define a new parameter "interrupts" as the following:</p> <p>PerfCounter = interrupts,"\\Processor(0)\\Interrupts/sec",60</p> <p>Please note double quotes around performance counter path.</p> <p>The parameter name (interrupts) is to be used as the item key when creating an item.</p> <p>Samples for calculating average value will be taken every second.</p> <p>You may run "typeperf -qx" to get list of all performance counters available in Windows.</p>

Parameter	Mandatory	Range	Default	Description
PerfCounterEn	no			<p>Defines a new parameter <parameter_name> which is an average value for system performance counter <perf_counter_path> for the specified time period <period> (in seconds). Syntax: <parameter_name>,"<perf_counter_path>",<period> Compared to PerfCounter, perfcounter paths must be in English. Supported only on Windows Server 2008/Vista and above. For example, if you wish to receive average number of processor interrupts per second for last minute, you can define a new parameter "interrupts" as the following: PerfCounterEn = interrupts,"\Processor(0)\Interrupts/sec",60 Please note double quotes around performance counter path. The parameter name (interrupts) is to be used as the item key when creating an item. Samples for calculating average value will be taken every second. You can find the list of English strings by viewing the following registry key: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\CurrentVersion\Perflib\009. This parameter is supported since Zabbix 4.0.13 and 4.2.7.</p>
RefreshActiveChecks	no	60-3600	120	<p>How often list of active checks is refreshed, in seconds. Note that after failing to refresh active checks the next refresh will be attempted after 60 seconds.</p>

Parameter	Mandatory	Range	Default	Description
Server	yes, if StartAgents is not explicitly set to 0			<p>List of comma delimited IP addresses, optionally in CIDR notation, or hostnames of Zabbix servers.</p> <p>Incoming connections will be accepted only from the hosts listed here.</p> <p>If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally and '::/0' will allow any IPv4 or IPv6 address.</p> <p>'0.0.0.0/0' can be used to allow any IPv4 address.</p> <p>Note, that "IPv4-compatible IPv6 addresses" (0000::/96 prefix) are supported but deprecated by RFC4291.</p> <p>Example: Server=127.0.0.1,192.168.1.0/24,::1,2001: Spaces are allowed.</p>

ServerActive	no	(*)	<p>List of comma-delimited IP address or DNS name (address:port) pairs or clusters (address:port;address2:port) of Zabbix servers and Zabbix proxies for active checks. Cluster nodes must be separated by a semicolon.</p> <p>Multiple comma-delimited addresses can be provided to use several independent Zabbix servers in parallel. Spaces are allowed.</p> <p>If port is not specified, default port is used.</p> <p>IPv6 addresses must be enclosed in square brackets if port for that host is specified.</p> <p>If port is not specified, square brackets for IPv6 addresses are optional.</p> <p>If this parameter is not specified, active checks are disabled.</p> <p>Example for multiple servers: ServerActive=127.0.0.1:20051,zabbix</p> <p>Example for high availability: ServerActive=zabbix.cluster.node1;zabbix.cluster.node2</p> <p>Example for high availability with two clusters and one server: ServerActive=zabbix.cluster.node1;zabbix.cluster.node2</p>
SourceIP	no		<p>Source IP address for:</p> <ul style="list-style-type: none"> - outgoing connections to Zabbix server or Zabbix proxy; - making connections while executing some items (web.page.get, net.tcp.port, etc.)

StartAgents	no	0-63 (*)	3	Number of pre-forked instances of zabbix_agentd that process passive checks. If set to 0, disables passive checks and the agent will not listen on any TCP port.
Timeout	no	1-30	3	Spend no more than Timeout seconds on processing
TLSAccept	yes, if TLS certificate or PSK parameters are defined (even for unencrypted connection), otherwise no			What incoming connections to accept. Used for a passive checks. Multiple values can be specified, separated by comma: unencrypted - accept connections without encryption (default) psk - accept connections with TLS and a pre-shared key (PSK) cert - accept connections with TLS and a certificate This parameter is supported since Zabbix 3.0.0.
TLSCAFile	no			Full pathname of a file containing the top-level CA(s) certificates for peer certificate verification, used for encrypted communications between Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSCertFile	no			Full pathname of a file containing the agent certificate or certificate chain, used for encrypted communications with Zabbix components. This parameter is supported since Zabbix 3.0.0.

TLSCConnect	yes, if TLS certificate or PSK parameters are defined (even for unencrypted connection), otherwise no	How the agent should connect to Zabbix server or proxy. Used for active checks. Only one value can be specified: unencrypted - connect without encryption (default) psk - connect using TLS and a pre-shared key (PSK) cert - connect using TLS and a certificate This parameter is supported since Zabbix 3.0.0.
TLSCRLFile	no	Full pathname of a file containing revoked certificates. This parameter is used for encrypted communications with Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSPKeyFile	no	Full pathname of a file containing the agent private key used for encrypted communications with Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSPSKFile	no	Full pathname of a file containing the agent pre-shared key used for encrypted communications with Zabbix components. This parameter is supported since Zabbix 3.0.0.
TLSPSKIdentity	no	Pre-shared key identity string, used for encrypted communications with Zabbix server. This parameter is supported since Zabbix 3.0.0.
TLSServerCertIssuer	no	Allowed server (proxy) certificate issuer. This parameter is supported since Zabbix 3.0.0.

TLSServerCertSubject	no			<p>Allowed server (proxy) certificate subject.</p> <p>This parameter is supported since Zabbix 3.0.0.</p>
UnsafeUserParameters	no	0-1	0	<p>Allow all characters to be passed in arguments to user-defined parameters.</p> <p>0 - do not allow 1 - allow</p> <p>The following characters are not allowed: \ ' " ' * ? [] { } ~ \$! & ; () > # @</p> <p>Additionally, newline characters are not allowed.</p>
UserParameter	no			<p>User-defined parameter to monitor. There can be several user-defined parameters.</p> <p>Format: UserParameter=<key>,<shell command></p> <p>Note that shell command must not return empty string or EOL only.</p> <p>Shell commands may have relative paths, if UserParameterDir parameter is specified.</p> <p>Examples: UserParameter=system.test,who wc -l</p>
UserParameterDir	no			<p>UserParameter=check_cpu,./custom.</p> <p>Default search path for UserParameter commands. If used, the agent will change its working directory to the one specified here before executing a command. Thereby, UserParameter commands can have a relative ./ prefix instead of a full path. Only one entry is allowed.</p> <p>Example: UserParameter-Dir=/opt/myscripts</p>

Note:

(*) The number of active servers listed in ServerActive plus the number of pre-forked instances for passive checks specified in StartAgents must be less than 64.

See also

1. [Differences in the Zabbix agent configuration for active and passive checks starting from version 2.0.0.](#)

6 Agente Zabbix 2 (Windows)

Visão geral

O Zabbix Agent 2 é uma nova geração de agente Zabbix e pode ser usado no lugar do Zabbix Agent.

Esta seção lista os parâmetros suportados em um arquivo de configuração do Zabbix Agent 2 (zabbix_agent2.win.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com “#” são suportados somente no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Alias	não			<p>Configura um apelido para uma chave de item. Ele pode ser usado para substituir uma chave de item longa e complexa por uma menor e mais simples. Múltiplos parâmetros Alias podem estar presentes. Múltiplos parâmetros com a mesma chave Alias são permitidos. Diferentes chaves Alias podem referenciar a mesma chave de item. Apelidos podem ser usados nos parâmetros HostMetadataItem mas não nos HostnameItem.</p> <p>Exemplos:</p> <p>1. Recuperando o ID do usuário 'zabbix'. Alias=zabbix.userid:vfs.file.regexp[/etc/passwd/9]+)"",\1] Agora a chave abreviada zabbix.userid pode ser usada para recuperação deste dado.</p> <p>2. Obtendo utilização de CPU com parâmetros padrão e customizados. Alias=cpu.util:system.cpu.util Alias=cpu.util[*]:system.cpu.util[*] Isto permite usar a chave cpu.util para obter a porcentagem de utilização de CPU com parâmetros padrão assim como usar cpu.util[all, idle, avg15] para obter dados específicos sobre utilização de CPU.</p> <p>3. Execução de múltiplas regras de descoberta de baixo-nível processando os mesmos itens descobertos. Alias=vfs.fs.discovery[*]:vfs.fs.discovery Agora é possível configurar várias regras de descoberta usando vfs.fs.discovery com parâmetros diferentes para cada regra, p.e., vfs.fs.discovery[foo], vfs.fs.discovery[bar], etc.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
AllowKey	não			<p>Permite execução das chaves de item que correspondem ao padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" para corresponder a qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com DenyKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
BufferSend	não	1-3600	5	<p>O intervalo de tempo em segundos que determina com que frequência os valores são enviados do buffer para o Zabbix Server. Note que se o buffer estiver cheio, os dados serão enviados mais cedo.</p>
BufferSize	não	2-65535	100	<p>Número máximo de valores em um buffer de memória. O agente enviará todos os dados coletados para o Zabbix Server ou Proxy se o buffer estiver cheio.</p> <p>Este parâmetro deve ser usado apenas se o buffer persistente estiver desabilitado (EnablePersistentBuffer=0).</p>
ControlSocket	não		\\.\pipe\agent.sock	<p>O socket de controle, usado para enviar comandos de tempo de execução com a opção '-R'.</p>
DebugLevel	não	0-5	3	<p>Especifica nível de depuração (debug):</p> <ul style="list-style-type: none"> 0 - informações básicas sobre início e parada dos processos do Zabbix 1 - informações críticas 2 - informações de erro 3 - alertas 4 - para depuração (produz bastante informação) 5 - depuração estendida (produz ainda mais informações)

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
DenyKey	não			<p>Nega a execução das chaves de item que correspondem ao padrão. Padrão de chave é uma expressão curinga que suporta o caracter "*" para corresponder a qualquer número de quaisquer caracteres.</p> <p>Múltiplas regras de correspondência de chave podem ser definidas em combinação com DenyKey. Os parâmetros são processados um por um de acordo com sua ordem de aparição.</p> <p>Este parâmetro é suportado desde o Zabbix 5.0.0.</p> <p>Veja também: Restringindo verificações do agente.</p>
EnablePersistentBuffer	não	0-1	0	<p>Habilita o uso de armazenamento local persistente para itens ativos.</p> <p>0 - desabilitado 1 - habilitado</p> <p>Se o armazenamento persistente estiver desabilitado, o buffer de memória será usado.</p>
HostInterface	não	0-255 caracteres		<p>Parâmetro opcional que define interface do host. A interface do host é usada no processo de autoregistro de host.</p> <p>Um agente emitirá um erro e não iniciará se o valor estiver acima do limite de 255 caracteres.</p> <p>Se não definido, o valor será adquirido de HostInterfaceltem.</p> <p>Suportado desde o Zabbix 4.4.0.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
HostInterfaceItem	não			<p>Parâmetro opcional que define um item usado para obtenção da interface do host.</p> <p>A interface do host é usada no processo de autoregistro de host.</p> <p>Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima do limite de 255 caracteres. Esta opção é usada apenas quando HostInterface não é definido.</p> <p>Suportado desde o Zabbix 4.4.0.</p>
HostMetadata	não	0-255 caracteres		<p>Parâmetro opcional que define os metadados do host. Os metadados de host são usados no processo de autoregistro.</p> <p>Um agente emitirá um erro e não iniciará se o valor especificado estiver acima do limite ou for uma string não-UTF-8.</p> <p>Se não definido, o valor será adquirido de HostMetadataItem.</p>
HostMetadataItem	não			<p>Parâmetro opcional que define um item usado para obtenção de metadados de host. O item de metadados de host é recuperado em cada tentativa de autoregistro para o processo de autoregistro de host.</p> <p>Durante uma requisição de autoregistro um agente registrará em log uma mensagem de alerta se o valor retornado pelo item especificado estiver acima do limite de 255 caracteres. Esta opção é usada apenas quando HostMetadata não é definido.</p> <p>Suporta UserParameters e apelidos (aliases). Suporta system.run[] independentemente do valor de EnableRemoteCommands.</p> <p>O valor retornado pelo item deve ser uma string UTF-8 caso contrário ele será ignorado.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Hostname	não		Configurado por HostnameItem	<p>Lista separada por vírgula de nomes de host únicos, sensíveis a maiúsculas e minúsculas.</p> <p>Necessário para verificações ativas e deve corresponder aos nomes de host conforme configurado no Server. O valor é adquirido de HostnameItem se não definido.</p> <p>Caracteres permitidos: alfanuméricos, '.', ',', '_', e '-'.</p> <p>Comprimento máximo: 128 caracteres por nome de host, 2048 caracteres para a linha inteira.</p>
HostnameItem	não		system.hostname	<p>Item usado para geração do Hostname se ele não estiver definido. Ignorado se Hostname estiver definido.</p> <p>Não suporta UserParameters ou apelidos, mas suporta system.run[] independentemente do valor de EnableRemoteCommands.</p> <p>O comprimento de saída é limitado em 512KB.</p>
Include	não			<p>Você pode incluir arquivos individuais ou todos os arquivos de um diretório no arquivo de configuração.</p> <p>Durante a instalação o Zabbix criará o diretório de include em /usr/local/etc, a menos que modificado durante o tempo de compilação.</p> <p>Para incluir apenas arquivos relevantes no diretório especificado, o caracter curinga asterisco é suportado para correspondência de padrão.</p> <p>Por exemplo: /absolute/path/to/config/files/*</p> <p>Desde o Zabbix 6.0.0 um caminho pode ser relativo à localização do arquivo zabbix_agent2.win.conf.</p> <p>Consulte notas especiais sobre limitações.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
ListenIP	não		0.0.0.0	Lista separada por vírgula de endereços IP que o agente deve ouvir. O primeiro endereço IP é enviado para o Zabbix Server, se conectando a ele, para recuperar a lista de verificações ativas.
ListenPort	não	1024-32767	10050	O agente ouvirá nesta porta por conexões vindas do Server.
LogFile	sim, se LogType estiver configurado como file, caso contrário não		c:\zabbix_agent2.log	Nome do arquivo de log se LogType for 'file'.
LogFileSize	não	0-1024	1	Tamanho máximo do arquivo de log em MB. 0 - desabilita rotação de log automática. Nota: Se o limite de tamanho de arquivo de log for alcançado e a rotação do arquivo falhar, por qualquer razão, o arquivo de log existente é truncado e iniciado outra vez.
LogType	não		file	Especifica onde as mensagens de log são gravadas: system - syslog, file - arquivo especificado pelo parâmetro LogFile, console - saída padrão.
PersistentBufferFile	não			O arquivo, onde o Zabbix Agent2 deve manter o banco de dados SQLite. Deve ser um nome de arquivo completo. Este parâmetro é usado apenas se o buffer persistente estiver habilitado (EnablePersistentBuffer=1).
PersistentBufferPeriod	não	1m-365d	1h	O período de tempo pelo qual o dado deve ser armazenado, quando não há conexão com o Server ou Proxy. Dados mais antigos serão perdidos. Dados de log serão preservados. Este parâmetro é usado somente se o buffer persistente estiver habilitado (EnablePersistentBuffer=1).
Plugins	não			Desde o Zabbix 6.0.0 a maioria dos plugins tem seus próprios arquivos de configuração. O arquivo de configuração contém os parâmetros de plugin listados abaixo.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
	Plugin não	log.MaxLinesPerSecond 1-1000	20	Número máximo de novas linhas que o agente enviará por segundo para o Zabbix Server ou Proxy quando processando verificações ativas de 'log' e 'eventlog'. O valor fornecido será sobrescrito pelo parâmetro 'maxlines', provido na chave de item 'log' ou 'eventlog'. Nota: O Zabbix processará 10 vezes mais novas linhas que o configurado em MaxLinesPerSecond para procurar a string requerida em itens de log. Este parâmetro é suportado desde o 4.4.2 e substitui MaxLinesPerSecond.
	Plugin não	system.run.log.remote.commands	0	Habilita registro de log de comandos shell executados como alertas. 0 - desabilitado 1 - habilitado Comandos serão registrados em log apenas se executados remotamente. Entradas de log não serão criadas se system.run[] for lançado localmente pelos parâmetros HostMetadataItem, HostInterfaceItem ou HostnameItem. Este parâmetro é suportado desde o 4.4.2 e substitui LogRemoteCommands.
RefreshActiveChecks	não	60-3600	120	Com que frequência a lista de verificações ativas é atualizada, em segundos. Note que após uma falha em atualizar as verificações ativas a próxima atualização será tentada após 60 segundos.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Server	yes			<p>Lista separada por vírgula de endereços IP, opcionalmente em notação CIDR, ou nomes DNS de Zabbix Servers e Zabbix Proxies.</p> <p>Conexões de entrada serão aceitas apenas a partir dos hosts listados aqui.</p> <p>Se o suporte a IPv6 estiver habilitado então '127.0.0.1', '::ffff:127.0.0.1' são tratados igualmente e '::/0' permitirão qualquer endereço IPv4 ou IPv6. '0.0.0.0/0' pode ser usado para permitir quaisquer endereços IPv4.</p> <p>Exemplo: Server=127.0.0.1,192.168.1.0/24,::1,2001</p>
ServerActive	não			<p>Espaços são permitidos.</p> <p>Lista separada por vírgula de pares IP:porta (ou nome DNS:porta) de Zabbix Servers e Zabbix Proxies para verificações ativas.</p> <p>Múltiplos endereços separados por vírgula podem ser fornecidos para usar vários Zabbix Servers independentes em paralelo.</p> <p>Espaços são permitidos.</p> <p>Se porta não for especificada, a porta padrão é usada.</p> <p>Endereços IPv6 devem ser envoltos por colchetes se a porta para o host for especificado.</p> <p>Se a porta não for especificada, os colchetes para endereços IPv6 são opcionais.</p> <p>Se este parâmetro não for especificado, verificações ativas são desabilitadas.</p> <p>Exemplo: ServerActive=127.0.0.1:20051,zabbix.example.com</p>
SourceIP	não			<p>Endereços IP de origem para:</p> <ul style="list-style-type: none"> - conexões de saída para o Zabbix Server ou Zabbix Proxy; - criação de conexões quando executando alguns itens (web.page.get, net.tcp.port, etc.)
StatusPort	não	1024-32767		<p>Se configurado, o agente ouvirá nesta porta por requisições de estado HTTP (http://localhost:<porta>/status).</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Timeout	não	1-30	3	Gastar não mais do que Timeout segundos no processamento.
TLSAccept	sim, se certificado TLS ou parâmetros PSK são definidos (mesmo para conexão não criptografada), caso contrário não			Quais conexões de entrada aceitar. Usado para verificações passivas. Múltiplos valores podem ser especificados, separados por vírgula: unencrypted - aceita conexões sem criptografia (padrão) psk - aceita conexões com TLS e uma chave pré-compartilhada (PSK) cert - aceita conexões com TLS e um certificado
TLSCAFile	não			Caminho completo de um arquivo contendo certificados CA(s) de topo-de-nível para verificação de pares de certificado, usado para comunicações criptografadas entre componentes Zabbix.
TLSCertFile	não			Caminho completo de um arquivo contendo os certificados de agente ou cadeia de certificado, usado para comunicações criptografadas com componentes Zabbix.
TLSConnect	sim, se certificado TLS ou parâmetros PSK são definidos (mesmo para conexão não criptografada), caso contrário não			Como o agente deve se conectar ao Zabbix Server ou Proxy. Usado para verificações ativas. Apenas um valor pode ser especificado: unencrypted - conecta sem criptografia (padrão) psk - conecta usando TLS e uma chave pré-compartilhada (PSK) cert - conecta usando TLS e um certificado
TLSCRLFile	não			Caminho completo de um arquivo contendo certificados revogados. Este parâmetro é usado para comunicações criptografadas com componentes Zabbix.
TLSKeyFile	não			Caminho completo de um arquivo contendo a chave privada do agente usada para comunicações criptografadas com componentes Zabbix.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
TLSPSKFile	não			Caminho completo de um arquivo contendo a chave pré-compartilhada do agente usada para comunicações criptografadas com componentes Zabbix.
TLSPSKIdentity	não			String de identidade de chave pré-compartilhada, usada para comunicações criptografadas com o Zabbix Server.
TLSServerCertIssuer	não			Emissor de certificado de Server (Proxy) permitido.
TLSServerCertSubject	não			Cliente de certificado de Server (Proxy) permitido.
UnsafeUserParameters	não	0,1	0	<p>Permite que todos os caracteres sejam passados em argumentos para parâmetros definidos por usuário.</p> <p>Os seguintes caracteres não são permitidos:</p> <pre>\ ' " * ? [] { } ~ \$! & ; () > # @</pre> <p>Adicionalmente, caracteres de nova linha não são permitidos.</p> <p>Parâmetro definido por usuário para monitoramento. Podem existir vários parâmetros definidos por usuário.</p> <p>Formato: UserParameter=<chave>,<comando shell></p> <p>Note que o comando shell não deve retornar uma string vazia ou apenas EOL. Comandos shell podem ter caminhos relativos, se o parâmetro UserParameterDir for especificado.</p> <p>Exemplos:</p> <pre>UserParameter=system.test,who wc -l</pre> <pre>UserParameter=check_cpu,./custom_script</pre>
UserParameter	não			

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
UserParameterDir	não			Caminho de busca padrão para comandos UserParameter. Se usado, o agente alterará seu diretório de operação para o especificado aqui antes da execução de um comando. Deste modo, comandos UserParameter podem ter um prefixo relativo ./ em vez de um caminho completo. Apenas uma entrada é permitida. Exemplo: UserParameter-Dir=/opt/myscripts

7 Plugins do Zabbix Agent 2

Visão geral

Esta seção contém descrições de parâmetros de arquivo de configuração para plugins do Zabbix Agent 2. Por favor utilize a barra lateral para acessar as informações sobre um plugin específico.

1 Plugin Ceph

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do plugin Ceph do Zabbix agent 2 (ceph.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Ceph.InsecureSkipVerify	não	falso / verdadeiro	falso	Determina se um cliente http deve verificar a cadeia de certificado e nome de servidor do Server. Se verdadeiro, o TLS aceita qualquer certificado apresentado pelo Server e qualquer nome de servidor no certificado. Neste modo, o TLS é suscetível a ataques "man-in-the-middle" (deveria ser usado apenas para testes).
Plugins.Ceph.KeepAlive	não	60-900	300	O tempo máximo de espera (em segundos) antes que conexões de plugins não usados sejam encerradas.
Plugins.Ceph.Sessions.<SessionName>.ApiKey	não			Chave de API da sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Ceph.Sessions.<SessionName>.User	não			Nome de usuário da sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item.
Plugins.Ceph.Sessions.<SessionName>.Uri	não		https://localhost:8003	String de conexão de uma sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item. Não deve incluir credenciais embutidas (elas serão ignoradas). Deve corresponder ao formato de URI. Apenas o esquema https é suportado; um esquema pode ser omitido (desde a versão 5.2.3). Uma porta pode ser omitida (padrão=8003). Exemplos: https://127.0.0.1:8003 localhost
Plugins.Ceph.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix agent 2 \(UNIX\)](#) / [Zabbix agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

2 Plugin Docker

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do Docker para o Zabbix Agent 2 (docker.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Docker.Endpoint	não		unix:///var/run/docker.sock	Localização do socket Unix do daemon Docker. Deve conter um esquema (apenas unix:// é suportado).
Plugins.Docker.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de **plugins**

3 Ember+ plugin

Visão Geral

Esta seção lista os parâmetros suportados no arquivo de configuração do plugin Ember+ Zabbix agent 2 (ember.conf).

O plugin Ember+ é um plugin carregável e está disponível e totalmente descrito no repositório [Ember+ plugin repository](#).

Este plugin é suportado desde o Zabbix 6.0.30 e atualmente só está disponível para ser compilado a partir do código-fonte (para Unix e Windows).

Observações:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração fornecidos;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem [BOM](#);
- Comentários que começam com "#" são suportados apenas no início da linha.

Parameters

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.EmberPlus.Default.Uri	Sim		tcp://localhost:9090	URI padrão para conexão. O único schema suportado é tcp://. Um schema pode ser omitido. Embedded credentials serão ignoradas.
Plugins.EmberPlus.KeepAlive	Sim	60-900	300	O tempo máximo de espera (em segundos) antes que as conexões de plugins não utilizadas sejam fechadas.
Plugins.EmberPlus.Sessions.<SessionName>.Uri	Sim		tcp://localhost:9090	URI para conexão, para a sessão nomeada. O único schema suportado é tcp://. Um schema pode ser omitido. Embedded credentials serão ignoradas. <SessionName> - define o nome de uma sessão para uso em chaves de item.
Plugins.EmberPlus.System.Path	Sim			Caminho para o executável do plugin Ember+. Exemplo de uso: <code>Plugins.EmberPlus.System.Path=/usr/sbin/zabbix-agent2-pl</code>
Plugins.EmberPlus.Timeout	Sim	1-30	timeout global	O tempo máximo de espera para uma resposta do servidor ao se conectar pela primeira vez e em operações de acompanhamento na sessão.

Consulte também:

- Descrição dos parâmetros gerais do Zabbix agent 2: [Zabbix agent 2 \(UNIX\)](#) / [Zabbix agent 2 \(Windows\)](#)
- Instrução para configuração dos **plugins**

3 Plugin Memcached

Overview

Esta seção lista os parâmetros suportados no arquivo de configuração do Memcached para o Zabbix Agent 2 (memcached.conf).

Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem [BOM](#);
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Memcached.KeepAlive	Sim	60-900	300	O tempo máximo de espera (em segundos) antes de encerrar a conexão de um plugin fora de uso.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Memcached.SessionName>.Password	sim			Senha da sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item.
Plugins.Memcached.SessionName>.Uri	sim		tcp://localhost:11211	String de conexão de uma sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item. Não deve incluir credenciais embutidas (elas serão ignoradas). Deve corresponder ao formato de URI. Esquemas suportados: tcp, unix; um esquema pode ser omitido (desde a versão 5.2.3). Uma porta pode ser omitida (padrão=11211). Exemplos: tcp://localhost:11211 localhost unix:/var/run/memcached.sock
Plugins.Memcached.SessionName>.User	sim			Nome de usuário da sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item.
Plugins.Memcached.Timeout	sim	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

4 Plugin Modbus

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do Modbus para o Zabbix Agent 2 (modbus.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Modbus.Sessions.<SessionName>.Endpoint	S			Endpoint é uma string de conexão consistindo em um esquema de protocolo, um endereço de host e uma porta ou nome de porta serial e atributos. <SessionName> - nome de uma sessão para uso em chaves de item.
Plugins.Modbus.Sessions.<SessionName>.SlaveID	S			ID secundário de uma sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item. Exemplo: Plugins.Modbus.Sessions.MB1.SlaveID Note que este parâmetro da sessão nomeada é verificado apenas se o valor fornecido no parâmetro ID secundário chave do item estiver vazio.
Plugins.Modbus.Sessions.<SessionName>.Timeout	S			Tempo limite de uma sessão nomeada. <SessionName> - nome de uma sessão para uso em chaves de item. Exemplo: Plugins.Modbus.Sessions.MB1.Timeout
Plugins.Modbus.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de **plugins**

5 Plugin MongoDB

Visão Geral

Esta seção lista os parâmetros suportados no arquivo de configuração do plugin do Zabbix Agent 2 para MongoDB (mongo.conf).

Desde o Zabbix 6.0.6, o plugin do MongoDB é um plugin carregável e está disponível e completamente descrito no [repositório do plugin MongoDB](#). [MongoDB plugin repository](#).

Binários pré-compilados do plugin para Windows estão disponíveis desde o [Zabbix 6.0.19](#) e são compatíveis com versões anteriores da 6.0.

Observações:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração enviados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem [BOM](#);
- Comentários que começam com **"#"** são suportados apenas no início da linha.

Opções

Parâmetro	Descrição
-V --version	Imprime a versão do plugin e informações de licença.
-h --help	Imprime informações de ajuda (forma abreviada).

Parâmetros

Note:

Nas versões do Zabbix anteriores à 6.0.6, os nomes dos parâmetros começam com Plugins.Mongo.<Parameter> em vez de Plugins.MongoDB.<Parameter>. Por exemplo, Plugins.Mongo.KeepAlive

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.MongoDB.Default.Password	Sim			Senha padrão para conectar ao MongoDB; usada se nenhum valor for especificado em uma chave de item ou sessão nomeada. Suportado desde a versão 6.0.18
Plugins.MongoDB.Default.Uri	Sim			URI padrão para conectar ao MongoDB; usada se nenhum valor for especificado em uma key de item ou sessão nomeada. Não deve incluir credenciais incorporadas (serão ignoradas). Deve corresponder ao formato de URI. Apenas o esquema tcp é suportado, um esquema pode ser omitido. Uma porta pode ser omitida (padrão=27017). Exemplos: tcp://127.0.0.1:27017, tcp:localhost, localhost Suportado desde a versão 6.0.18
Plugins.MongoDB.Default.User	Sim			Nome de usuário padrão para conectar ao MongoDB; usado se nenhum valor for especificado em uma key de item ou sessão nomeada. Suportado desde a versão 6.0.18
Plugins.MongoDB.KeepAlive	Sim	60-900	300	O tempo máximo de espera (em segundos) antes que as conexões de plugin não utilizadas sejam fechadas.
Plugins.MongoDB.Sessions.<SessionName>.Password	Sim			Senha da sessão nomeada. <SessionName> - define o nome de uma sessão para uso em keys de item..
Plugins.MongoDB.Sessions.<SessionName>.TLSCAFile	Sim			Nome completo do arquivo contendo os certificados CA (Certificado de Autoridade) de nível superior para verificação do certificado do par, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados. <SessionName> - define o nome de uma sessão para uso em keys de item..
Plugins.MongoDB.Sessions.<SessionName>.TLSCertFile	Sim			Nome completo do arquivo contendo o certificado do agente Zabbix 2, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados. <SessionName> - define o nome de uma sessão para uso em keys de item..
Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile	Sim			Nome completo do arquivo contendo a chave privada do agente Zabbix 2, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados. <SessionName> - define o nome de uma sessão para uso em keys de item..
Plugins.MongoDB.Sessions.<SessionName>.TLSConnectfor	Sim			Nome completo do arquivo contendo o certificado do agente Zabbix 2, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados. <SessionName> - define o nome de uma sessão para uso em keys de item..

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.MongoDB.Sessions.<SessionName>.TLSConnect	Sim			<p>Tipo de criptografia para comunicações entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em keys de item..</p> <p>Valores suportados: required - requer conexão TLS; verify_ca - verificar certificados;; verify_full - verificar certificados e endereço IP.</p> <p>Suportado nas versões do plugin 1.2.0, 6.0.13 e mais recentes¹.</p>
Plugins.MongoDB.Sessions.<SessionName>.TLSKeyFile	Sim			<p>Nome completo do arquivo contendo a chave privada do banco de dados usada para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em keys de item.</p> <p>Antes da versão 6.0.24 sempre obrigatório se Plugins.MongoDB.Sessions.<SessionName>.TLSConnect estiver configurado para um dos: verify_ca, verify_full</p> <p>Suportado nas versões do plugin 1.2.0, 6.0.13 e mais recentes¹.</p>
Plugins.MongoDB.Sessions.<SessionName>.Uri	Sim			

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
String de conexão de uma sessão nomeada. <SessionName> - define o nome de uma sessão para uso em keys de item.				
Não deve incluir credenciais incorporadas (serão ignoradas). Deve corresponder ao formato de URI format. Apenas o esquema tcp é suportado; um esquema pode ser omitido. Uma porta pode ser omitida (padrão=27017). Exemplos: tcp://127.0.0.1:27017, tcp:localhost, localhost				
Plugins.MongoDBSessions.<SessionName>.Usuário				Nome do usuário da sessão nomeada. <SessionName> - define o nome de uma sessão para uso em key de item.
Plugins.MongoDBSystem.Path				Caminho para o executável do plugin externo. Suportado desde o Zabbix 6.0.6
Plugins.MongoDBTimeout		1-30	tempo limite global	Tempo limite da execução (quanto tempo esperar para que uma solicitação seja concluída antes de encerrá-la).

Consulte também:

- Descrição dos parâmetros de configuração do gerais do agente [Zabbix agent 2 \(UNIX\) / Zabbix agent 2 \(Windows\)](#)
- Instruções para configurar [plugins](#)

Notas de rodapé

¹ - Desde o Zabbix 6.0.13, os plugins carregáveis passaram a usar o mesmo sistema de versionamento que o próprio Zabbix. Como resultado, a versão do plugin do MongoDB mudou de 1.2.0 para 6.0.13.

6 Plugin MQTT

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do MQTT para o Zabbix Agent 2 (mqtt.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;

- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem [BOM](#);
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.MQTT.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

7 Plugin MSSQL

Visão Geral

Esta seção lista os parâmetros suportados no arquivo de configuração do plugin MSSQL Zabbix agent 2 (mssql.conf).

Este plugin é suportado desde o Zabbix 6.0.27. Para mais informações, consulte o readme [MSSQL plugin](#).

Observe que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração enviados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem [BOM](#);
- Comentários que começam com "#" são suportados apenas no início da linha.

Parameters

Parameter	Mandatory	Range	Default	Description
Plugins.MSSQL.CustomQueriesDir			empty	Specifies the file path to a directory containing user-defined .sql files with custom queries that the plugin can execute. The plugin loads all available .sql files in the configured directory at startup. This means that any changes to the custom query files will not be reflected until the plugin is restarted. The plugin is started and stopped together with Zabbix agent 2.
Plugins.MSSQL.Default.CACertPath				The default file path to the public key certificate of the certificate authority (CA) that issued the certificate of the MSSQL server. The certificate must be in PEM format.
Plugins.MSSQL.Default.Database				The default database name to connect to.
Plugins.MSSQL.Default.Encrypt				Specifies the default connection encryption type. Possible values are: true - data sending between plugin and server is encrypted; false - data sending between plugin and server is not encrypted beyond the login packet; strict - data sending between plugin and server is encrypted E2E using TDS8 ; disable - data sending between plugin and server is not encrypted.
Plugins.MSSQL.Default.HostNameInCertificate				The common name (CN) of the certificate of the MSSQL server by default.
Plugins.MSSQL.Default.Password				The password to be sent to a protected MSSQL server by default.
Plugins.MSSQL.Default.TLSMinVersion				The minimum TLS version to use by default. Possible values are: 1.0, 1.1, 1.2, 1.3.
Plugins.MSSQL.Default.TrustServerCertificate				Whether the plugin should trust the server certificate without validating it by default. Possible values: true, false.
Plugins.MSSQL.Default.Uri			sqlserver://localhost	The default TDS URI to connect. The only supported schema is sqlserver://. A schema can be omitted. Embedded credentials will be ignored.

Parameter	Mandatory	Range	Default	Description
Plugins.MSSQL.Default.User	Yes			The default username to be sent to a protected MSSQL server.
Plugins.MSSQL.KeepAlive	Yes	60-900	300	The maximum time of waiting (in seconds) before unused plugin connections are closed.
Plugins.MSSQL.Sessions.<SessionName>.CACertPath	Yes			The file path to the public key certificate of the certificate authority (CA) that issued the certificate of the MSSQL server for the named session. The certificate must be in PEM format. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.Database	Yes			The database name to connect to for the named session. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.Encrypt	Yes			Specifies the connection encryption type for the named session. Possible values are: true - data sending between plugin and server is encrypted; false - data sending between plugin and server is not encrypted beyond the login packet; strict - data sending between plugin and server is encrypted E2E using TDS8 ; disable - data sending between plugin and server is not encrypted. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.HostNameInCertificate	Yes			The common name (CN) of the certificate of the MSSQL server for the named session. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.Password	Yes			The password to be sent to a protected MSSQL server for the named session. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.TLSMinVersion	Yes			The minimum TLS version to use for the named session. Possible values are: 1.0, 1.1, 1.2, 1.3. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.TrustServerCertificate	Yes			Whether the plugin should trust the server certificate without validating it for the named session. Possible values: true, false. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.Uri	Yes		sqlserver://localhost	The URI to connect, for the named session. The only supported schema is <code>sqlserver://</code> . A schema can be omitted. Embedded credentials will be ignored. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.Sessions.<SessionName>.User	Yes			The username to be sent to a protected MSSQL server for the named session. <SessionName> - define name of a session for using in item keys.
Plugins.MSSQL.System.Path	Yes			Path to the MSSQL plugin executable. Global setting for the MSSQL plugin. Applied to all connections. Example usage: <code>Plugins.MSSQL.System.Path=/usr/sbin/zabbix-agent2-plugin</code>
Plugins.MSSQL.Timeout	No	1-30	global timeout	The amount of time to wait for a server to respond when first connecting and on follow-up operations in the session.

See also:

- Description of general Zabbix agent 2 configuration parameters: [Zabbix agent 2 \(UNIX\) / Zabbix agent 2 \(Windows\)](#)
- Instructions for configuring [plugins](#)

8 Plugin MySQL

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do MySQL para o Zabbix Agent 2 (mysql.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com “#” são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Mysql.Default.Timeout	Sim	1-30	timeout global	O tempo máximo em segundos para esperar que uma solicitação seja concluída.
Plugins.Mysql.Default.CustomQueriesPath	Sim		vazio	Caminho completo para o diretório usado para armazenar consultas personalizadas. Suportado desde a versão 6.0.21.
Plugins.Mysql.Default.Default.Senha	Sim			Senha padrão para conectar-se ao MySQL; usada se nenhum valor for especificado em uma chave de item ou sessão nomeada. Suportado desde a versão 6.0.18.
Plugins.Mysql.Default.Default.TLSCAFile	Sim, se Plugins.Mysql.Default.TLSConnect estiver definido como verify_ca ou verify_full)			Nome completo do arquivo contendo o(s) certificado(s) da CA de nível superior para verificação de certificado do par para comunicações criptografadas entre o agente Zabbix 2 e bancos de dados monitorados; usado se nenhum valor for especificado em uma sessão nomeada. Suportado desde a versão 6.0.18.
Plugins.Mysql.Default.Default.TLSCertFile	Sim, se Plugins.Mysql.Default.TLSConnect estiver definido para verify_ca ou verify_full)			Nome completo do arquivo contendo o certificado do agente ou cadeia de certificados para comunicações criptografadas entre o agente Zabbix 2 e bancos de dados monitorados; usado se nenhum valor for especificado em uma sessão nomeada. Suportado desde a versão 6.0.18.
Plugins.Mysql.Default.Default.TLSConnect	Sim			Tipo de criptografia para comunicações entre o agente Zabbix 2 e bancos de dados monitorados; usado se nenhum valor for especificado em uma sessão nomeada. Valores suportados: required - requer conexão TLS; verify_ca - verificar certificados; verify_full - verificar certificados e endereço IP. Suportado desde a versão 6.0.18.
Plugins.Mysql.Default.Default.TLSKeyFile	Sim, se Plugins.Mysql.Default.TLSConnect estiver definido para verify_ca ou verify_full)			Nome completo do arquivo contendo a chave privada do banco de dados para comunicações criptografadas entre o agente Zabbix 2 e bancos de dados monitorados; usado se nenhum valor for especificado em uma sessão nomeada. Suportado desde a versão 6.0.18.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Mysql.Default.Uri	Sim		tcp://localhost:3306	<p>Não padrão para conectar-se ao MySQL; usado se nenhum valor for especificado em uma chave de item ou sessão nomeada.</p> <p>Não deve incluir embedded credentials (elas serão ignoradas). Deve corresponder ao formato de URI.</p> <p>Schemes suportados: tcp, unix; um scheme pode ser omitido.</p> <p>Uma porta pode ser omitida (padrão=3306).</p> <p>Exemplos: tcp://localhost:3306 localhost unix:/var/run/mysql.sock</p> <p>Suportado desde a versão 6.0.18.</p>
Plugins.Mysql.Default.User	Sim			<p>Nome de usuário padrão para conectar-se ao MySQL; usado se nenhum valor for especificado em uma chave de item ou sessão nomeada.</p> <p>Suportado desde a versão 6.0.18.</p>
Plugins.Mysql.KeepAlive	Sim	60-900	300	<p>O tempo máximo de espera (em segundos) antes que conexões de plugin não utilizadas sejam fechadas.</p>
Plugins.Mysql.Sessions.<SessionName>.Senha	Sim			<p>Senha da sessão nomeada.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSCAFile	Sim			<p>Nome completo do arquivo contendo o(s) certificado(s) da CA de nível superior para verificação de certificado, usado para comunicações criptografadas entre o agente Zabbix 2 e bancos de dados monitorados.</p> <p><SessionName> - Define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSConnect	Sim			<p>Nome completo de um arquivo que contém o certificado do agente ou cadeia de certificados, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	Sim			<p>Nome completo de um arquivo que contém o certificado do agente ou cadeia de certificados, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	Sim			<p>Nome completo de um arquivo que contém o certificado do agente ou cadeia de certificados, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	Sim			<p>Nome completo de um arquivo que contém o certificado do agente ou cadeia de certificados, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	Sim			<p>Nome completo de um arquivo que contém o certificado do agente ou cadeia de certificados, usado para comunicações criptografadas entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p>
Plugins.Mysql.Sessions.<SessionName>.TLSConnect	Sim			<p>O tipo de criptografia para comunicações entre o agente Zabbix 2 e os bancos de dados monitorados.</p> <p><SessionName> - define o nome de uma sessão para uso em chaves de item.</p> <p>Valores suportados: required - requer conexão TLS; verify_ca - verifica certificados; verify_full - verifica certificados e endereços de IP.</p>

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Mysql.Sessions.Plugins.Mysql.Sessions.<SessionName>.TLSKeyFile	Sim			Caminho completo de um arquivo contendo a chave privada do banco de dados usada para comunicações criptografadas entre o Zabbix Agent 2 e os bancos de dados monitorados.
Plugins.Mysql.Sessions.Plugins.Mysql.Sessions.<SessionName>.TLSCertFile	Sim			<SessionName> - define o nome de uma sessão para uso em chaves de item.
Plugins.Mysql.Sessions.Plugins.Mysql.Sessions.<SessionName>.TLSConnect	Antes da versão 6.0.23, sempre obrigatório se Plugins.Mysql.Sessions.<SessionName>.TLSCertFile estiver definido para verify_ca ou verify_full			
Plugins.Mysql.Sessions.Plugins.Mysql.Sessions.<SessionName>.Uri	Sim			String de conexão de uma sessão nomeada. <SessionName> - define o nome de uma sessão para uso em chaves de item.
Plugins.Mysql.Sessions.Plugins.Mysql.Sessions.<SessionName>.User	Sim			Não deve incluir embedded credentials (elas serão ignoradas). Deve corresponder ao formato de URI. Schemes suportados: tcp, unix; um scheme pode ser suportado. Uma porta pode ser suportada (padrão=3306). Exemplos: tcp://localhost:3306 localhost unix:/var/run/mysql.sock
Plugins.Mysql.Timeout	Sim	1-30	timeout global	Tempo limite de execução da solicitação (quanto tempo esperar para que uma solicitação seja concluída antes de encerrá-la).

Consulte também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix agent 2 \(UNIX\)](#) / [Zabbix agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

8 Plugin Oracle

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do Ceph para o Zabbix Agent 2 (ceph.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Oracle.CallTimeout	não	1-30	tempo limite global	Tempo de espera máximo em segundos para uma requisição estar completa.
Plugins.Oracle.ConnectTimeout	sim	1-30	tempo limite global	Tempo de espera máximo em segundos para uma conexão ser estabelecida.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Oracle.CustomQueriesPath	sim			Caminho completo de um diretório contendo arquivos .sql com consultas customizadas. Desabilitado por padrão. Exemplo: /etc/zabbix/oracle/sql
Plugins.Oracle.KeepAlive	não	60-900	300	O tempo máximo de espera (em segundos) antes de encerrar conexões de plugin não utilizadas.
Plugins.Oracle.Sessions.<SessionName>.Password	sim			Senha da sessão nomeada. <SessionName> - nome de uma sessão para uso nas chaves de item.
Plugins.Oracle.Sessions.<SessionName>.Service	sim			Nome do serviço da sessão nomeada para ser usado para conexão (SID não é suportado). Suportado para: Oracle. <PluginName> - nome do plugin. <SessionName> - nome de uma sessão para uso nas chaves de item.
Plugins.Oracle.Sessions.<SessionName>.Uri	sim		tcp://localhost:1521	String de conexão da sessão nomeada para Oracle. <SessionName> - nome de uma sessão para uso nas chaves de item. Não deve incluir credenciais embutidas (elas serão ignoradas). Deve corresponder ao formato de URI. Apenas o esquema tcp é suportado; um esquema pode ser omitido (desde a versão 5.2.3). Uma porta pode ser omitida (padrão=1521). Exemplos: tcp://127.0.0.1:1521 localhost
Plugins.Oracle.Sessions.<SessionName>.User	sim			Nome de usuário da sessão nomeada. <SessionName> - nome de uma sessão para uso nas chaves de item.

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\) / Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

9 Puglin do PostgreSQL

Visão geral

Essa seção lista os parâmetros suportados no arquivo de configuração do plugin do agente PostgreSQL Zabbix 2 (postgresql.conf).

Desde o Zabbix 6.0.10, o PostgreSQL é um plugin carregável, que está disponível e descrito por completo em [PostgreSQL plugin repository](#)

Note que:

- Os valores padrão refletem o padrão do processo, e não os valores nos arquivos de configuração fornecidos;
- O Zabbix suporta somente arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários que começam com "#" somente são suportados no início da linha.

Opções

Parâmetro	Descrição
-V --version	Exibe a versão do plugin e informações sobre licença.
-h --help	Exibe informação de ajuda (atalho).

Parâmetros

Note:

Nas versões anteriores de Zabbix 6.0.10, os nomes de parâmetro começavam com Plugins.Postgres.<Parameter> ao invés de Plugins.PostgreSQL.<Parameter>. Por exemplo, Plugins.Postgres.KeepAlive.

Parâmetro	Obrigatório	Range	Default	Description
Plugins.PostgreSQL.CallTimeout	1-30		global timeout	Maximum wait time (in seconds) for a request to be completed.
Plugins.PostgreSQL.CustomQueriesPath			disabled	Full pathname of the directory containing .sql files with custom queries.
Plugins.PostgreSQL.Default.Database				Default database for connecting to PostgreSQL; used if no value is specified in an item key or named session. Supported since version 6.0.18.
Plugins.PostgreSQL.Default.Password				Default password for connecting to PostgreSQL; used if no value is specified in an item key or named session. Supported since version 6.0.18.
Plugins.PostgreSQL.Default.TLSCAFile (sim, se Plugins.PostgreSQL.Default.TLSConnect estiver configurado para: verify_ca, verify_full)				Full pathname of a file containing the top-level CA(s) certificate for peer certificate verification for encrypted communications between Zabbix agent 2 and monitored databases; used if no value is specified in a named session. Supported since version 6.0.18.
Plugins.PostgreSQL.Default.TLSCertFile (sim, se Plugins.PostgreSQL.Default.TLSConnect estiver configurado para: verify_ca, verify_full)				Full pathname of a file containing the PostgreSQL certificate or certificate chain for encrypted communications between Zabbix agent 2 and monitored databases; used if no value is specified in a named session. Supported since version 6.0.18.
Plugins.PostgreSQL.Default.TLSConnect				Encryption type for communications between Zabbix agent 2 and monitored databases; used if no value is specified in a named session. Supported values: required - connect using TLS as transport mode without identity checks; verify_ca - connect using TLS and verify certificate; verify_full - connect using TLS, verify certificate and verify that database identity (CN) specified by DBHost matches its certificate. Undefined encryption type means unencrypted connection. Supported since version 6.0.18.

Parâmetro	Obrigatório	Range	Default	Description
Plugins.PostgreSQL.Default.TLSKeyFile	Yes			Full pathname of a file containing the PostgreSQL private key for encrypted communications between Zabbix agent 2 and monitored databases; used if no value is specified in a named session. Supported since version 6.0.18.
Plugins.PostgreSQL.Default.TLSConnect	Yes			
Plugins.PostgreSQL.Default.Uri	Yes			Default URI for connecting to PostgreSQL; used if no value is specified in an item key or named session. Should not include embedded credentials (they will be ignored). Must match the URI format. Supported schemes: <code>tcp</code> , <code>unix</code> . Examples: <code>tcp://127.0.0.1:5432</code> <code>tcp://localhost</code> <code>unix:/var/run/postgresql/.s.PGSQL.5432</code> Supported since version 6.0.18.
Plugins.PostgreSQL.Default.User	Yes			Default username for connecting to PostgreSQL; used if no value is specified in an item key or named session. Supported since version 6.0.18.
Plugins.PostgreSQL.KeepAlive	Yes	60-900	300	Maximum time of waiting (in seconds) before unused plugin connections are closed.
Plugins.PostgreSQL.Sessions.<SessionName>.Database	Yes			Database for session connection. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.Sessions.<SessionName>.Password	Yes			Password for session connection. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCAFile	Yes			Full pathname of a file containing the top-level CA(s) certificate peer certificate verification. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSCertFile	Yes			Full pathname of a file containing the PostgreSQL certificate or certificate chain. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect	Yes			Encryption type for PostgreSQL connection. <SessionName> - define name of a session for using in item keys. Supported values: required - connect using TLS as transport mode without identity checks; verify_ca - connect using TLS and verify certificate; verify_full - connect using TLS, verify certificate and verify that database identity (CN) specified by DBHost matches its certificate. Undefined encryption type means unencrypted connection.

Parâmetro	Obrigatório	Range	Default	Description
Plugins.PostgreSQL.Sessions.<SessionName>.TLSKeyFile	Sim, se Plugins.PostgreSQL.Sessions.<SessionName>.TLSConnect estiver configurado para: verify_ca, verify_full)			Full pathname of a file containing the PostgreSQL private key. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.Sessions.<SessionName>.Uri				Connection string of a named session. <SessionName> - define name of a session for using in item keys. Should not include embedded credentials (they will be ignored). Must match the URI format. Supported schemes: tcp, unix. Examples: tcp://127.0.0.1:5432 tcp://localhost unix:/var/run/postgresql/.s.PGSQL.5432
Plugins.PostgreSQL.Sessions.<SessionName>.User				Named session username. <SessionName> - define name of a session for using in item keys.
Plugins.PostgreSQL.System.Path				Path to external plugin executable. Supported since Zabbix 6.0.10.
Plugins.PostgreSQL.Timeout		1-30	global timeout	Request execution timeout (how long to wait for a request to complete before shutting it down).

Veja também:

- Descrição geral dos parâmetros de configuração do agente Zabbix 2: [Zabbix agent 2 \(UNIX\) / Zabbix agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

10 Plugin Redis

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do Redis para o Zabbix Agent 2 (redis.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Redis.KeepAlive	não	60-900	300	O tempo máximo de espera (em segundos) antes de encerrar conexões de plugin não utilizadas.
Plugins.Redis.Sessions.<SessionName>.Password	Sim			Senha da sessão nomeada. <SessionName> - nome de uma sessão para uso nas chaves de item.

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Redis.Sessions.<SessionName>.Uri	não		tcp://localhost:6379	String de conexão de uma sessão nomeada. <SessionName> - nome de uma sessão para uso nas chaves de item. Não deve incluir credenciais embutidas (elas serão ignoradas). Deve corresponder ao formato de URI. Esquemas suportados: tcp, unix; um esquema pode ser omitido (desde a versão 5.2.3). Uma porta pode ser omitida (padrão=6379). Exemplos: tcp://localhost:6379 localhost unix:/var/run/redis.sock
Plugins.Redis.Sessions.<SessionName>.User	não			Nome de usuário da sessão nomeada. <SessionName> - nome de uma sessão para uso nas chaves de item.
Plugins.Redis.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)

11 Plugin Smart

Visão geral

Esta seção lista os parâmetros suportados no arquivo de configuração do Smart para o Zabbix Agent 2 (smart.conf). Note que:

- Os valores padrão refletem os padrões do processo, não os valores nos arquivos de configuração transportados;
- O Zabbix suporta apenas arquivos de configuração codificados em UTF-8 sem BOM;
- Comentários começando com "#" são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Intervalo	Padrão	Descrição
Plugins.Smart.Path	não		smartctl	Caminho para o executável smartctl.
Plugins.Smart.Timeout	não	1-30	tempo limite global	Tempo limite de execução da requisição (quanto esperar pela conclusão de uma requisição antes de encerrá-la).

Veja também:

- Descrição dos parâmetros de configuração gerais do Zabbix Agent 2: [Zabbix Agent 2 \(UNIX\)](#) / [Zabbix Agent 2 \(Windows\)](#)
- Instruções para configuração de [plugins](#)(/manual/config/items/plugin)

8 Zabbix Java Gateway

Se você usar os scripts `startup.sh` e `shutdown.sh` para iniciar o **Zabbix Java Gateway**, então você pode especificar os parâmetros de configuração necessários no arquivo `settings.sh`. Os scripts de início e parada carregam o arquivo de configurações e cuidam em converter variáveis shell (listadas na primeira coluna) para propriedades Java (listadas na segunda coluna).

Se você iniciar o Zabbix Java Gateway manualmente pela execução do `java` diretamente, então você especifica as propriedades Java correspondentes na linha de comando.

Variável	Propriedade	Obrigatório	Intervalo	Padrão	Descrição
LISTEN_IP	<code>zabbix.listenIP</code>	não		0.0.0.0	Endereço IP no qual aguardar comunicação.
LISTEN_PORT	<code>zabbix.listenPort</code>	não	1024-32767	10052	Porta na qual ouvir.
PID_FILE	<code>zabbix.pidFile</code>	não		<code>/tmp/zabbix_java.pid</code>	Nome do arquivo PID. Se omitido, o Zabbix Java Gateway é iniciado como uma aplicação de console.
PROPERTIES_FILE	<code>zabbix.propertiesFile</code>	não			Nome do arquivo de propriedades. Pode ser usado para definir propriedades adicionais usando um formato chave-valor de forma que elas não são visíveis em uma linha de comando ou para sobrescrever os existentes. Por exemplo: "javax.net.ssl.trustStorePassword=<p>
START_POLLERS	<code>zabbix.startPollers</code>	não	1-1000	5	Número de threads de operadores a iniciar.
TIMEOUT	<code>zabbix.timeout</code>	não	1-30	3	Quanto esperar por operações de rede.

Warning:

A porta 10052 não é [registrada na IANA](#).

9 Serviço web Zabbix

Visão geral

O serviço web Zabbix é um processo que é usado para comunicação com serviços web externos.

Esta seção lista os parâmetros suportados no serviço web Zabbix arquivo de configuração (`zabbix_web_service.conf`). Observe que:

- Os valores padrão refletem os padrões do processo, não os valores no arquivos de configuração enviados;
- Zabbix suporta arquivos de configuração apenas em codificação UTF-8 sem BOM;
- Comentários começando com `"#"` são suportados apenas no início da linha.

Parâmetros

Parâmetro	Obrigatório	Faixa	Padrão	Descrição
AllowedIP	sim			<p>Lista de endereços IP delimitados por vírgulas, opcionalmente em notação CIDR, ou nomes DNS de servidores Zabbix e proxies Zabbix.</p> <p>As conexões de entrada serão aceitas apenas dos hosts listados aqui.</p> <p>Se o suporte IPv6 estiver ativado, então 127.0.0.1, ::127.0.0.1, ::ffff:127.0.0.1 são tratados igualmente e ::/0 permitirá qualquer endereço IPv4 ou IPv6.</p> <p>0.0.0.0/0 pode ser usado para permitir qualquer endereço IPv4.</p> <p>Exemplo: 127.0.0.1, 192.168.1.0/24, ::1, 2001::</p>
DebugLevel	não	0-5	3	<p>Especifica o nível de depuração:</p> <p>0 - informações básicas sobre início e parada de processos Zabbix</p> <p>1 - informações críticas</p> <p>2 - informações de erro</p> <p>3 - avisos</p> <p>4 - para depuração (produz muitas informações)</p> <p>5 - depuração estendida (produz ainda mais informações)</p>
ListenPort	não	1024-32767	10053	O serviço de porta de conexões com o servidor.
LogFile	sim, se LogType estiver definido como file, caso contrário, não			<p>Nome do arquivo de log para o parâmetro 'file' LogType.</p> <p>Exemplo: /tmp/zabbix_web_service.log</p>
LogFileSize	não	0-1024	1	<p>Tamanho máximo do arquivo de log em MB.</p> <p>0 - desativar rotação automática de logs.</p>
LogType	não	sistemas/ arquivos/ console	file	<p>Especifica onde as mensagens de log são gravadas:</p> <p>system - syslog</p> <p>file - arquivo especificado com o parâmetro LogFile</p> <p>console - saída padrão</p>
Timeout	não	1-30	3	Não gaste mais de segundos de tempo limite no processamento.

Parâmetro	Obrigatório	Faixa	Padrão	Descrição
TLSAccept	não	não criptografado/certificado	não criptografado Especifica que tipo de conexão usar: unencrypted - aceita conexões sem criptografia (padrão) cert - aceita conexões com TLS e certificados	
TLSCAFile	não			Nome do caminho completo de um arquivo contendo os certificados de CA(s) de nível superior para verificação de certificado de peer, usado para comunicações criptografadas entre componentes do Zabbix.
TLSCertFile	não			Caminho completo de um arquivo contendo o certificado de serviço ou cadeia de certificados, usado para comunicações criptografadas com componentes Zabbix.
TLSKeyFile	não			Caminho completo de um arquivo contendo a chave privada de serviço usada para comunicações criptografadas com componentes Zabbix.

10 Inclusão

Visão geral

Arquivos adicionais ou diretórios podem ser incluídos na configuração do Server/Proxy/Agent usando o parâmetro `Include`.

Notas sobre inclusão

Se o parâmetro `Include` é usado para a inclusão de um arquivo, o arquivo deve ser legível.

Se o parâmetro `Include` é usado para inclusão de um diretório:

- Todos os arquivos no diretório devem ser legíveis.
- Nenhuma ordem de inclusão em particular deve ser assumida (p.e. arquivos não são incluídos em ordem alfabética).
- Todos os arquivos no diretório são incluídos na configuração.
- Atente-se para cópias backup de arquivos criadas automaticamente por alguns editores de texto. Por exemplo, arquivos com nomes como `file.backup` não são incluídos.

Se o parâmetro `Include` é usado para inclusão de arquivos usando um padrão:

- Todos os arquivos correspondentes ao padrão devem ser legíveis.
- Nenhuma ordem de inclusão em particular deve ser assumida (p.e. arquivos não são incluídos em ordem alfabética).

4 Protocolos

1 Protocolo de troca de dados servidor-proxy

Visão geral

Servidor - a troca de dados do proxy é baseada no formato JSON.

As mensagens de solicitação e resposta devem começar com **cabeçalho e dados comprimento**.

Proxy passivo

Solicitação de configuração de proxy

A solicitação proxy config é enviada pelo servidor para fornecer proxy dados de configuração. Esta solicitação é enviada a cada ProxyConfigFrequency (parâmetro de configuração do servidor) segundos.

nome	tipo de valor	descrição
servidor→proxy:		
request	string	'proxy config'
<tabela>	objeto	um ou mais objetos com <tabela> dados
campos	array	array de nomes de campo
-	string	nome do campo
dados	array	array de linhas
-	array	array de colunas
-	string,number	valor da coluna com tipo dependendo do tipo de coluna no esquema do banco de dados
proxy→servidor:		
resposta	string	as informações de sucesso da solicitação ('sucesso' ou 'falha')
versão	string	a versão do proxy (<major>.<minor>.<build>)

Exemplo:

servidor→proxy:

```
{
  "request": "configuração de proxy",
  "macro global":{
    "Campos":[
      "globalmacróide",
      "macro",
      "valor"
    ],
    "dados":[
      [
        2,
        "{$SNMP_COMMUNITY}",
        "público"
      ]
    ]
  },
  "hosts":{
    "Campos":[
      "host",
      "hospedeiro",
      "status",
      "ipmi_authtype",
      "ipmi_privilege",
      "ipmi_username",
      "ipmi_senha",
      "nome",
      "tls_connect",
      "tls_aceitar",
      "tls_issuer",
      "tls_assunto",
      "tls_psk_identity",
      "tls_psk"
    ],
    "dados": [
```

```

[
    10001,
    "Linux",
    3,
    -1,
    2,
    "",
    "",
    "Linux",
    1,
    1,
    "",
    "",
    "",
    ""
],
[
    10050,
    "Agente Zabbix",
    3,
    -1,
    2,
    "",
    "",
    "Agente Zabbix",
    1,
    1,
    "",
    "",
    "",
    ""
],
[
    10105,
    "Registrador",
    0,
    -1,
    2,
    "",
    "",
    "Registrador",
    1,
    1,
    "",
    "",
    "",
    ""
]
]
},
"interface":{
    "Campos":[
        "interfaceid",
        "host",
        "a Principal",
        "tipo",
        "usei",
        "ip",
        "dns",
        "porta",
        "volume"
    ],

```

```

    "dados": [
      [
        2,
        10105,
        1,
        1,
        1,
        "127.0.0.1",
        "",
        "10050",
        1
      ]
    ]
  },
  ...
}

```

proxy→servidor:

```

{
  "resposta": "sucesso",
  "versão": "5.4.0"
}

```

Solicitação de proxy

A solicitação proxy data é usada para obter a disponibilidade da interface do host, dados históricos, de descoberta e registro automático do proxy. Esse pedido é enviado a cada ProxyDataFrequency (parâmetro de configuração do servidor) segundos.

nome	tipo de valor	descrição
servidor→proxy: solicitação	string	'dados de proxy'
proxy→servidor: sessão	string	token de sessão de dados
disponibilidade de interface	array	(opcional) array de objetos de dados de disponibilidade de interface
interface_id	identificador de interface	identificador de interface
disponível	Disponibilidade da interface	Disponibilidade da interface
		0 , INTERFACE_AVAILABLE_UNKNOWN - desconhecido
		1 , *INTERFACE_AVAILABLE_TRUE* - disponível
		2 , INTERFACE_AVAILABLE_FALSE - indisponível
dados históricos	error string array	Mensagem de erro de interface ou string vazia (opcional) array de objetos de dados históricos
	item_id número	identificador do item
	relógio número	valor do item timestamp (segundos)
	ns número	valor do item timestamp (nanossegundos)
	valor string	(opcional) valor do item
	id number	identificador de valor (contador crescente, único dentro de uma sessão de dados)
	timestamp number	(opcional) timestamp de itens do tipo de log
	fonte string	(opcional) valor da fonte do item do log de eventos
	severidade number	(opcional) valor de gravidade do item do log de eventos
	event_id number	(opcional) eventlog item eventid value
	state string	(opcional) estado do item
		0 , ITEM_STATE_NORMAL
		1 , *ITEM_STATE_NÃO SUPORTADO*
	last_log_size number	(opcional) último tamanho de log dos itens do tipo de log

nome	tipo de valor	descrição
dados de descoberta	mtime number	(opcional) modificar a hora dos itens do tipo de log
	array	(opcional) array de objetos de dados de descoberta
	relógio número	o carimbo de data/hora dos dados de descoberta
	ruleid number	o identificador da regra de descoberta
	dcheckid number	o identificador de verificação de descoberta ou nulo para dados da regra de descoberta
	tipo número	tipo de verificação de descoberta:
		-1 dados da regra de descoberta
		0, SVC_SSH - serviço SSH check
		1, SVC_LDAP - verificação do serviço LDAP
		2, SVC_SMTP - verificação do serviço SMTP
		3, SVC_FTP - Verificação do serviço FTP
		4, SVC_HTTP - Verificação do serviço HTTP
		5, SVC_POP - Verificação do serviço POP
		6, SVC_NNTP - Verificação do serviço NNTP
		7, SVC_IMAP - Verificação do serviço IMAP
		8, SVC_TCP - Verificação da disponibilidade da porta TCP
		* *9, SVC_AGENT - agente Zabbix 10, SVC_SNMPv1 - agente SNMPv1 11, SVC_SNMPv2 - agente SNMPv2 12, SVC_ICMPPING - ping ICMP 13, SVC_SNMPv3 - agente SNMPv3 14, SVC_HTTPS - verificação de serviço HTTPS 15*, SVC_TELNET - Verificação de disponibilidade do Telnet
	ip string	o endereço IP do host
	dns string	o nome DNS do host
	porta número	(opcional) número da porta de serviço
	key_ string	(opcional) a chave do item para verificação de descoberta do tipo 9 SVC_AGENT
	valor string	(opcional) valor recebido do serviço, pode estar vazio para a maioria dos serviços
	status number	(opcional) status do serviço:
		0, DOBJECT_STATUS_UP - Service UP
		**1*, DOBJECT_STATUS_DOWN* - Serviço DOWN
	registro automático	(opcional) array de objetos de dados de registro automático
	relógio número	o carimbo de data/hora dos dados de registro automático
	host string	o nome do host
	ip string	(opcional) o endereço IP do host
	dns string	(opcional) o nome DNS resolvido do endereço IP
	port string	(opcional) a porta do host
	host_metadata	(opcional) os metadados do host enviados pelo agente (com base no parâmetro de configuração do agente HostMetadata ou HostMetadataItem)
	tarefas	(opcional) array de tarefas
	type number	o tipo de tarefa:
		0,
		ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT
		- remote command result

nome	tipo de valor	descrição
status	number	status de execução do comando remoto:
		0 , ZBX_TM_REMOTE_COMMAND_COMPLETED - o comando remoto foi concluído com sucesso 1 , ZBX_TM_REMOTE_COMMAND_FAILED - o comando remoto falhou (opcional) a mensagem de erro parentTaskId o ID da tarefa pai (opcional) 1 - há mais dados do histórico para enviar relógio (opcional) data de transferência de dados (segundos) ns (opcional) data de transferência de dados (nanossegundos) versão a versão do proxy (<major>.<minor>.<build>) servidor→proxy: resposta as informações de sucesso da solicitação ('sucesso' ou 'falha') tarefas (opcional) array de tarefas tipo tipo de tarefa: 1 , ZBX_TM_TASK_PROCESS_REMOTE_COMMAND - comando remoto hora de criação da tarefa relógio o tempo em segundos após o qual a tarefa expira ttd commandType o tipo de comando remoto: 0 1 , ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT - use script personalizado 2 , ZBX_SCRIPT_TYPE_IPMI - use IPMI 3 , ZBX_SCRIPT_TYPE_SSH - use SSH 4 , ZBX_SCRIPT_TYPE_TELNET - use Telnet 5 , ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT - use script global (atualmente funcionalmente equivalente ao script personalizado) command o comando remoto a ser executado executeOn o destino de execução para scripts personalizados: 0 , ZBX_SCRIPT_EXECUTE_ON_AGENT - executa script no agente 1 , ZBX_SCRIPT_EXECUTE_ON_SERVER - executa script no servidor 2 , ZBX_SCRIPT_EXECUTE_ON_PROXY - executa script no proxy

nome	tipo de valor	descrição
port	number	(opcional) a porta para comandos telnet e ssh
authtype	number	(opcional) o tipo de autenticação para comandos ssh
username	string	(opcional) o nome de usuário para comandos telnet e ssh
senha	string	(opcional) a senha para comandos telnet e ssh
chave pública	string	(opcional) a chave pública para comandos ssh
privatekey	string	(opcional) a chave privada para comandos ssh
parenttaskid	number	o ID da tarefa pai
hostid	number	hostid de destino

Exemplo:

servidor→proxy:

```
{
  "request": "dados proxy"
}
```

proxy→servidor:

```
{
  "sessão": "12345678901234567890123456789012"
  "disponibilidade de interface": [
    {
      "interfaceid": 1,
      "disponível": 1,
      "erro": ""
    },
    {
      "interfaceid": 2,
      "disponível": 2,
      "error": "Falha ao obter valor do agente: não é possível conectar a [[127.0.0.1]:10049]: [111]"
    },
    {
      "interfaceid": 3,
      "disponível": 1,
      "erro": ""
    },
    {
      "interfaceid": 4,
      "disponível": 1,
      "erro": ""
    }
  ],
  "dados do histórico": [
    {
      "itemid": "12345",
      "relógio": 1478609647,
      "ns": 332510044,
      "valor": "52956612",
      "id": 1
    },
    {
      "itemid": "12346",
      "relógio": 1478609647,
      "ns": 330690279,
      "estado": 1,
      "value": "Não foi possível encontrar informações para esta interface de rede em /proc/net/dev.",
      "id": 2
    }
  ]
}
```



```

],
"dados de descoberta":[
  {
    "relógio": 1478608764,
    "drule": 2,
    "dcheck":3,
    "tipo":12,
    "ip":"10.3.0.10",
    "dns":"vdebian",
    "estado":1
  },
  {
    "relógio": 1478608764,
    "drule": 2,
    "dcheck":null,
    "tipo 1,
    "ip":"10.3.0.10",
    "dns":"vdebian",
    "estado":1
  }
],
"registro automático":[
  {
    "relógio":1478608371,
    "host":"Registrador1",
    "ip":"10.3.0.1",
    "dns":"localhost",
    "porta":"10050"
  },
  {
    "relógio":1478608381,
    "host":"Logger2",
    "ip":"10.3.0.2",
    "dns":"localhost",
    "porta":"10050"
  }
],
"tarefas":[
  {
    "tipo": 0,
    "estado": 0,
    "parent_taskid": 10
  },
  {
    "tipo": 0,
    "estado": 1,
    "error": "Nenhuma permissão para executar a tarefa.",
    "parent_taskid": 20
  }
],
"versão":"5.4.0"
}

```

servidor→proxy:

```

{
  "resposta": "sucesso",
  "tarefas":[
    {
      "tipo 1,
      "relógio": 1478608371,
      "ttl": 600,
      "commandtype": 2,

```

```

        "command": "restart_service1.sh",
        "execute_on": 2,
        "porta": 80,
        "authtype": 0,
        "username": "userA",
        "senha": "senha1",
        "publickey": "MIGfMAOGCSqGSib3DQEBAQUAA4GNADCBiQKBgQCqGKuk01De7zhZj6+H0qtjTkVxwTCpvKe",
        "privatekey": "lsuusFncCzWBQ7RKNUsesmQRMSGkVb1/3j+skZ6UtW+5u091HNsj6tQ5QCqGKuk01De7zhd",
        "parent_taskid": 10,
        "hostid": 10070
    },
    {
        "tipo": 1,
        "relógio": 1478608381,
        "ttl": 600,
        "commandtype": 1,
        "command": "restart_service2.sh",
        "execute_on": 0,
        "authtype": 0,
        "nome do usuário": "",
        "senha": "",
        "chave pública": "",
        "chave privada": "",
        "parent_taskid": 20,
        "hostid": 10084
    }
]
}

```

Proxy ativo

Solicitação de request de proxy

A solicitação proxy heartbeat é enviada pelo proxy para informar que o proxy está executando. Este pedido é enviado a cada HeartbeatFrequency (proxy parâmetro de configuração) segundos.

nome	tipo de valor	descrição
proxy→servidor:		
request	string	'proxy heartbeat'
host	string	o nome do proxy
versão	string	a versão do proxy (<major>.<minor>.<build>)
servidor→proxy:		
resposta	string	as informações de sucesso da solicitação ('sucesso' ou 'falha')

proxy→servidor:

```

{
  "request": "proxy heartbeat",
  "host": "Proxy nº 12",
  "versão": "5.4.0"
}

```

servidor→proxy:

```

{
  "resposta": "sucesso"
}

```

Solicitação de configuração de proxy

A solicitação proxy config é enviada pelo proxy para obter o proxy dados de configuração. Este pedido é enviado a cada ConfigFrequency (proxy parâmetro de configuração) segundos.

nome	tipo de valor	descrição
proxy→servidor:		
request	string	'proxy config'
host	string	nome do proxy
versão	string	a versão do proxy (<major>.<minor>.<build>)
servidor→proxy:		
request	string	'proxy config'
<tabela>	objeto	um ou mais objetos com <tabela> dados
campos	array	array de nomes de campo
-	string	nome do campo
dados	array	array de linhas
-	array	array de colunas
-	string,number	valor da coluna com tipo dependendo do tipo de coluna no esquema do banco de dados
proxy→servidor:		
resposta	string	as informações de sucesso da solicitação ('sucesso' ou 'falha')

Exemplo:

proxy→servidor:

```
{
  "request": "configuração de proxy",
  "host": "Proxy nº 12",
  "versão": "5.4.0"
}
```

servidor→proxy:

```
{
  "macro global":{
    "Campos": [
      "globalmacróide",
      "macro",
      "valor"
    ],
    "dados": [
      [
        2,
        "{$SNMP_COMMUNITY}",
        "público"
      ]
    ]
  },
  "hosts":{
    "Campos": [
      "host",
      "hospedeiro",
      "status",
      "ipmi_authtype",
      "ipmi_privilege",
      "ipmi_username",
      "ipmi_senha",
      "nome",
      "tls_connect",
      "tls_aceitar",
      "tls_issuer",
      "tls_assunto",
      "tls_psk_identity",
      "tls_psk"
    ]
  }
}
```

```

    ],
    "dados": [
        [
            10001,
            "Linux",
            3,
            -1,
            2,
            "",
            "",
            "Linux",
            1,
            1,
            "",
            "",
            "",
            ""
        ],
        [
            10050,
            "Agente Zabbix",
            3,
            -1,
            2,
            "",
            "",
            "Agente Zabbix",
            1,
            1,
            "",
            "",
            "",
            ""
        ],
        [
            10105,
            "Registrador",
            0,
            -1,
            2,
            "",
            "",
            "Registrador",
            1,
            1,
            "",
            "",
            "",
            ""
        ]
    ]
},
"interface": {
    "Campos": [
        "interfaceid",
        "host",
        "a Principal",
        "tipo",
        "usei",
        "ip",
        "dns",
        "porta",

```

```

        "volume"
    ],
    "dados": [
        [
            2,
            10105,
            1,
            1,
            1,
            "127.0.0.1",
            "",
            "10050",
            1
        ]
    ]
},
...
}

```

proxy→servidor:

```

{
  "resposta": "sucesso"
}

```

Solicitação de dados de proxy

A solicitação proxy data é enviada pelo proxy para fornecer a interface do host dados de disponibilidade, histórico, descoberta e registro automático. Esse pedido é enviado a cada DataSenderFrequency (parâmetro de configuração de proxy) segundos.

nome	tipo de valor	descrição
proxy→servidor:		
solicitação	string	'dados de proxy'
host	string	o nome do proxy
sessão	string	token de sessão de dados
disponibilidade de interface	array	(opcional) array de objetos de dados de disponibilidade de interface
	interfaceid número	identificador de interface
	disponível número	Disponibilidade da interface
		0 , INTERFACE_AVAILABLE_UNKNOWN - desconhecido
		1 , *INTERFACE_AVAILABLE_TRUE* - disponível
		2 , INTERFACE_AVAILABLE_FALSE - indisponível
dados	error string	Mensagem de erro de interface ou string vazia
históricos	array	(opcional) array de objetos de dados históricos
	itemid número	identificador do item
	relógio número	valor do item timestamp (segundos)
	ns número	valor do item timestamp (nanossegundos)
	valor string	(opcional) valor do item
	id number	identificador de valor (contador crescente, único dentro de uma sessão de dados)
	timestamp number	(opcional) timestamp de itens do tipo de log
	fonte string	(opcional) valor da fonte do item do log de eventos
	severidade número	(opcional) valor de gravidade do item do log de eventos
	eventid number	(opcional) eventlog item eventid value
	state string	(opcional) estado do item
		0 , ITEM_STATE_NORMAL
		1 , *ITEM_STATE_NÃO SUPORTADO*
	lastlogsize number	(opcional) último tamanho de log dos itens do tipo de log

nome	tipo de valor	descrição
dados de descoberta	mtime number	(opcional) modificar a hora dos itens do tipo de log
	array	(opcional) array de objetos de dados de descoberta
	relógio número	o carimbo de data/hora dos dados de descoberta
	ruleid number	o identificador da regra de descoberta
	dcheckid number	o identificador de verificação de descoberta ou nulo para dados da regra de descoberta
	tipo número	tipo de verificação de descoberta:
		-1 dados da regra de descoberta
		0, SVC_SSH - serviço SSH check
		1, SVC_LDAP - verificação do serviço LDAP
		2, SVC_SMTP - verificação do serviço SMTP
		3, SVC_FTP - Verificação do serviço FTP
		4, SVC_HTTP - Verificação do serviço HTTP
		5, SVC_POP - Verificação do serviço POP
		6, SVC_NNTP - Verificação do serviço NNTP
		7, SVC_IMAP - Verificação do serviço IMAP
		8, SVC_TCP - Verificação da disponibilidade da porta TCP
		* *9, SVC_AGENT - agente Zabbix 10, SVC_SNMPv1 - agente SNMPv1 11, SVC_SNMPv2 - agente SNMPv2 12, SVC_ICMPPING - ping ICMP 13, SVC_SNMPv3 - agente SNMPv3 14, SVC_HTTPS - verificação de serviço HTTPS 15*, SVC_TELNET - Verificação de disponibilidade do Telnet
	ip string	o endereço IP do host
	dns string	o nome DNS do host
	porta número	(opcional) número da porta de serviço
	key_ string	(opcional) a chave do item para verificação de descoberta do tipo 9 SVC_AGENT
	valor string	(opcional) valor recebido do serviço, pode estar vazio para a maioria dos serviços
	status number	(opcional) status do serviço:
		0, DOBJECT_STATUS_UP - Service UP
		**1*, DOBJECT_STATUS_DOWN* - Serviço DOWN
	registro automático	(opcional) array de objetos de dados de registro automático
	relógio número	o carimbo de data/hora dos dados de registro automático
	host string	o nome do host
	ip string	(opcional) o endereço IP do host
	dns string	(opcional) o nome DNS resolvido do endereço IP
	port string	(opcional) a porta do host
	host_metadata	(opcional) os metadados do host enviados pelo agente (com base no parâmetro de configuração do agente HostMetadata ou HostMetadataItem)
	tarefas	(opcional) array de tarefas
	type number	o tipo de tarefa:
		0,
		ZBX_TM_TASK_PROCESS_REMOTE_COMMAND_RESULT
		- remote command result

nome	tipo de valor	descrição
status	number	o status de execução do comando remoto:
		0 , ZBX_TM_REMOTE_COMMAND_COMPLETED - o comando remoto foi concluído com sucesso 1 , ZBX_TM_REMOTE_COMMAND_FAILED - o comando remoto falhou (opcional) a mensagem de erro
error	string	
parent_task_id	number	o ID da tarefa pai
mais	número	(opcional) 1 - há mais dados do histórico para enviar
relógio	número	(opcional) data de transferência de dados (segundos)
ns	number	(opcional) data de transferência de dados (nanossegundos)
versão	string	a versão do proxy (<major>.<minor>.<build>)
servidor→proxy:		
resposta	string	as informações de sucesso da solicitação ('sucesso' ou 'falha')
tarefas	array	(opcional) array de tarefas
tipo	número	tipo de tarefa:
		1 , ZBX_TM_TASK_PROCESS_REMOTE_COMMAND - comando remoto hora de criação da tarefa o tempo em segundos após o qual a tarefa expira o comando remoto type:
relógio	número	
ttl	number	
tipo de comando	número	0 , ZBX_SCRIPT_TYPE_CUSTOM_SCRIPT - usar script personalizado 1 , ZBX_SCRIPT_TYPE_IPMI - usar IPMI 2 , ZBX_SCRIPT_TYPE_SSH - usar SSH 3 , ZBX_SCRIPT_TYPE_TELNET - usar Telnet 4 , ZBX_SCRIPT_TYPE_GLOBAL_SCRIPT - usa script global (atualmente funcionalmente equivalente ao script personalizado) o comando remoto a ser executado o destino de execução para scripts personalizados:
comando	string	
execute_on	number	0 , ZBX_SCRIPT_EXECUTE_ON_AGENT - executa script no agente 1 , ZBX_SCRIPT_EXECUTE_ON_SERVER - executa script no servidor 2 , ZBX_SCRIPT_EXECUTE_ON_PROXY - executa script no proxy (opcional) a porta para comandos telnet e ssh (opcional) o tipo de autenticação para comandos ssh (opcional) o nome de usuário para comandos telnet e ssh (opcional) a senha para comandos telnet e ssh (opcional) a chave pública para comandos ssh (opcional) a chave privada para comandos ssh
port	number	
auth_type	number	
username	string	
senha	string	
chave pública	string	
private_key	string	
parent_task_id	number	o ID da tarefa pai
hostid	number	hostid de destino

Exemplo:

proxy→servidor:

```
{
  "request": "proxy data",
  "host": "Proxy #12",
  "session": "12345678901234567890123456789012",
  "interface availability": [
    {
      "interfaceid": 1,
      "available": 1,
      "error": ""
    },
    {
      "interfaceid": 2,
      "available": 2,
      "error": "Get value from agent failed: cannot connect to [[127.0.0.1]:10049]: [111] Connection
    },
    {
      "interfaceid": 3,
      "available": 1,
      "error": ""
    },
    {
      "interfaceid": 4,
      "available": 1,
      "error": ""
    }
  ],
  "history data":[
    {
      "itemid":"12345",
      "clock":1478609647,
      "ns":332510044,
      "value":"52956612",
      "id": 1
    },
    {
      "itemid":"12346",
      "clock":1478609647,
      "ns":330690279,
      "state":1,
      "value":"Cannot find information for this network interface in /proc/net/dev.",
      "id": 2
    }
  ],
  "discovery data":[
    {
      "clock":1478608764,
      "drule":2,
      "dcheck":3,
      "type":12,
      "ip":"10.3.0.10",
      "dns":"vdebian",
      "status":1
    },
    {
      "clock":1478608764,
      "drule":2,
      "dcheck":null,
      "type":-1,
      "ip":"10.3.0.10",
      "dns":"vdebian",
      "status":1
    }
  ]
}
```



```

    }
],
"auto_registration":[
    {
        "clock":1478608371,
        "host":"Logger1",
        "ip":"10.3.0.1",
        "dns":"localhost",
        "port":"10050"
    },
    {
        "clock":1478608381,
        "host":"Logger2",
        "ip":"10.3.0.2",
        "dns":"localhost",
        "port":"10050"
    }
],
"tasks":[
    {
        "type": 2,
        "clock":1478608371,
        "ttl": 600,
        "commandtype": 2,
        "command": "restart_service1.sh",
        "execute_on": 2,
        "port": 80,
        "authtype": 0,
        "username": "userA",
        "password": "password1",
        "publickey": "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCqGKuk01De7zhZj6+H0qtjTkVxwTCpvKe",
        "privatekey": "lsuusFncCzWBQ7RKNUSesmQRMSGkVb1/3j+skZ6UtW+5u09lHNsj6tQ5QCqGKuk01De7zhd",
        "parent_taskid": 10,
        "hostid": 10070
    },
    {
        "type": 2,
        "clock":1478608381,
        "ttl": 600,
        "commandtype": 1,
        "command": "restart_service2.sh",
        "execute_on": 0,
        "authtype": 0,
        "username": "",
        "password": "",
        "publickey": "",
        "privatekey": "",
        "parent_taskid": 20,
        "hostid": 10084
    }
],
"tasks":[
    {
        "type": 0,
        "status": 0,
        "parent_taskid": 10
    },
    {
        "type": 0,
        "status": 1,
        "error": "No permissions to execute task.",
        "parent_taskid": 20
    }
]

```

```

    }
  ],
  "version": "5.4.0"
}

```

servidor→proxy:

```

{
  "response": "success",
  "upload": "enabled",
  "tasks": [
    {
      "type": 1,
      "clock": 1478608371,
      "ttl": 600,
      "commandtype": 2,
      "command": "restart_service1.sh",
      "execute_on": 2,
      "port": 80,
      "authtype": 0,
      "username": "userA",
      "password": "password1",
      "publickey": "MIGfMAOGCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCqGKuk01De7zhZj6+H0qtjTkVxwTCpvKe",
      "privatekey": "lsuusFncCzWBQ7RKNUSesmQRMSGkVb1/3j+skZ6UtW+5u091HNsj6tQ5QCqGKuk01De7zhd",
      "parent_taskid": 10,
      "hostid": 10070
    },
    {
      "type": 1,
      "clock": 1478608381,
      "ttl": 600,
      "commandtype": 1,
      "command": "restart_service2.sh",
      "execute_on": 0,
      "authtype": 0,
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "parent_taskid": 20,
      "hostid": 10084
    }
  ]
}

```

2 Protocolo do agente Zabbix

Consulte [Agente passivo e ativo checks](#) para mais informações.

3 Protocolo de envio Zabbix

Por favor, consulte a página de [trapper item](#) Para maiores informações.

3 Protocolo do agente Zabbix 2

Visão geral

Esta seção fornece informações sobre:

- Agent2 -> Server : ativar solicitação de verificações
- Server -> Agent2 : ativar resposta de verificações

- Agent2 -> Server : solicitação de dados do agente
- Server -> Agent2 : resposta de dados do agente

Solicitação de verificações ativas

A solicitação de verificações ativas é usada para obter as verificações ativas a serem processadas pelo agente. Esta solicitação é enviada pelo agente ao iniciar e, em seguida, em intervalos com RefreshActiveChecks.

Campo	Tipo	Obrigatório	Valor
request	string	sim	active checks
host	string	sim	Nome host.
version	string	sim	Versão do agente: <major>.<minor>.
host_metadata	string	não	Parâmetro de configuração HostMetadata ou HostMetadataItem valor métrico.
interface	string	não	Parâmetro de configuração HostInterface ou HostInterfaceItem valor métrico.
ip	string	não	Parâmetro de configuração ListenIP first IP se configurado.
port	number	não	Parâmetro de configuração valor ListenPort se configurado and não agente padrão listening port.

Exemplo:

```
{
  "request": "active checks",
  "host": "Zabbix server",
  "version": "6.0",
  "host_metadata": "mysql,nginx",
  "hostinterface": "zabbix.server.lan",
  "ip": "159.168.1.1",
  "port": 12050
}
```

Respostas das verificações ativas

A resposta das verificações ativas é enviada pelo servidor de volta ao agente depois de processar a solicitação de verificações ativas.

Campo	Tipo	Obrigatório	Valor
response	string	sim	success failed
info	string	não	Informações de erro no caso de falha.
data	array of objects	não	Itens de verificação ativa.
key	string	não	Item key com macros expandidas.
itemid	number	não	Identificador do item.
delay	string	não	Intervalo de atualização do item.
lastlogsize	number	não	Item lastlogsize.
mtime	number	não	Item mtime.
regex	array of objects	não	Global regular expressions.
name	string	não	Global regular expression name.
expression	string	não	Expressão regular global.
expression_type	number	não	Tipo Expressão regular global.
exp_delimiter	string	não	Delimitador Expressão regular global.
case_sensitive	number	não	Configuração de sensibilidade a maiúsculas/minúsculas para expressão regular global.

Exemplo:

```
{
  "response": "success",
  "data": [
    {
      "key": "log[/home/zabbix/logs/zabbix_agentd.log]",
      "itemid": 1234,

```

```

    "delay": "30s",
    "lastlogsize": 0,
    "mtime": 0
  },
  {
    "key": "agent.version",
    "itemid": 5678,
    "delay": "10m",
    "lastlogsize": 0,
    "mtime": 0
  }
]
}

```

Solicitação de dados do agente

A solicitação de dados do agente contém valores dos itens coletados.

Field	Tipo	Obrigatório	Valor
request	string	sim	agent data
host	string	sim	Nome do Host.
version	string	sim	A versão do agente: <maior>.<memnor>.
session	string	sim	O identificador de sessão único é gerado cada vez que um agente é iniciado.
data	array de objetos	sim	Valores de itens.
id	número	sim	Identificador de valor (contador incremental usado para verificar valores duplicados no caso de problemas de rede).
itemid	número	sim	Identificador de item.
value	string	não	O valor de item.
lastlogsize	número	não	O item lastlogsize.
mtime	número	não	O item mtime.
state	número	não	O status do item.
source	string	não	O valor do evento de origem do registro.
eventid	número	não	O valor do ID do registro de eventos.
severity	número	não	O valor da severidade do registro de eventos.
timestamp	número	não	O valor do carimbo de data/hora do registro de eventos.
clock	número	sim	O valor do carimbo de data/hora (segundos desde Epoch).
ns	número	sim	O valor do carimbo de data/hora em nanosegundos.

Exemplo:

```

{
  "request": "agent data",
  "data": [
    {
      "id": 1,
      "itemid": 5678,
      "value": "2.4.0",
      "clock": 1400675595,
      "ns": 76808644
    },
    {
      "id": 2,
      "itemid": 1234,
      "lastlogsize": 112,
      "value": " 19845:20140621:141708.521 Starting Zabbix Agent [<hostname>]. Zabbix 2.4.0 (revision 5000)",
      "clock": 1400675595,
      "ns": 77053975
    }
  ],
  "host": "Zabbix server",

```

```

"version": "6.0",
"session": "1234456akdsjhfoi"
}

```

Resposta de dados do agente

A resposta de dados do agente é enviada pelo servidor de volta para o agente após o processamento de solicitação de dados do agente.

Field	Tipo	Obrigatório	Valor
response	string	sim	success failed
info	string	sim	Resultados do processamento do item.

Exemplo:

```

{
  "response": "success",
  "info": "processed: 2; failed: 0; total: 2; seconds spent: 0.003534"
}

```

4 Cabeçalho

Visão geral

O cabeçalho está presente nas mensagens de resposta e solicitação entre o Zabbix componentes. É necessário determinar o comprimento da mensagem, se for compactado ou não e o formato dos campos de comprimento da mensagem. O cabeçalho consiste em:

<PROTOCOLO> - "ZBXD" (4 bytes).

<FLAGS> - os sinalizadores de protocolo, (1 byte). 0x01 - protocolo de comunicação Zabbix, 0x02 - compress

<DATALEN> - comprimento dos dados (4 bytes ou 8 bytes para pacotes grandes). 1 será formatado como 01/00/0

<RESERVED> - comprimento de dados não compactados (4 bytes ou 8 bytes para pacotes grandes). 1 será format

Quando a compactação está habilitada (sinalizador 0x02) os bytes <RESERVED> contém tamanho de dados não compactado.

O protocolo Zabbix tem um limite de tamanho de pacote de 1 GB por conexão. O limite de 1 GB é aplicado para comprimento de dados de pacote recebido e para dados não compactados comprimento, no entanto, quando o pacote grande está habilitado (flag 0x04) é possível para que o proxy Zabbix receba configuração com tamanho de até 16GB; Nota esse pacote grande só pode ser usado para configuração de proxy Zabbix, e O servidor Zabbix definirá automaticamente (flag 0x04) e enviará campos de comprimento como 8 bytes cada quando o comprimento dos dados antes da compactação exceder 4 GB.

Structure

The header consists of four fields. All numbers in the header are formatted as little-endian.

Field	Size	Size (large packet)	Description
<PROTOCOL>	4	4	"ZBXD" or 5A 42 58 44
<FLAGS>	1	1	Protocol flags: 0x01 - Zabbix communications protocol 0x02 - compression 0x04 - large packet
<DATALEN>	4	8	Data length.
<RESERVED>	4	8	When compression is used (0x02 flag) - the length of uncompressed data When compression is not used - 00 00 00 00

Implementação

Aqui estão trechos de código mostrando como adicionar o cabeçalho do protocolo Zabbix ao data que você deseja enviar para obter o pacote que você deveria enviar para o Zabbix para que seja interpretado corretamente.

Idioma	Código
bash	<code>printf -v LENGTH '%016x' "\${#DATA}"PACK=</code>
Java	<code>byte[] cabeçalho = new byte[] { 'Z', 'B',</code>
PHP	<code>\$packet = "ZBXD\1" . pack('P', strlen(\$d</code>
Perl	<code>my \$packet = "ZBXD\1" . pack('<Q', compr</code>
Python	<code>packet = "ZBXD\1" + struct.pack('<Q', le</code>

4 Protocolo do plugin no agente Zabbix 2

O protocolo do Zabbix agente 2 é baseado em código, tamanho e modelo de dados.

Código

Tipo	Tamanho	Comentários
Byte	4	Tipo do Payload, atualmente somente o JSON é suportado.

Tamanho

Tipo	Tamanho	Comentários
Byte	4	Tamanho do Payload atual em bytes.

Dado Payload

Tipo	Tamanho	Comentários
Byte	Definido pelo campo Size	Dados formatados em JSON.

Definição dados Payload

Dado comum

Estes parâmetros são apresentados em todos requests/responses:

Nome	Tipo	Comentários
id	uint32	Para requests - o indicador crescente usado para vincular requests com responses. Único dentro de uma request direta (i.e. do agente para o plugin ou do plugin para o agente).
type	uint32	Para responses - ID da request correspondente. O tipo request.

Request Log

A request enviada por um plugin para escrever a mensagem de log no arquivo de log do agente. | | |---|---| |direction|plugin → agent| |response|no|

Parâmetros específicos para requests log:

Nome	Tipo	Comentários
severity	uint32	A gravidade da mensagem (nível log).
message	string	A mensagem para o log.

Exemplo:

```
{"id":0,"type":1,"severity":3,"message":"message"}
```

Request registro

Um request enviada pelo agente durante a fase de inicialização do agente para obter métricas fornecidas para registrar um plugin.

direction	agent → plugin
response	yes

Parâmetros específicos para solicitação de registro:

Nome	Tipo	Comentários
version	string	Versão do protocolo <major>.<minor>

Exemplo:

```
{"id":1,"type":2,"version":"1.0"}
```

Resposta de registro

A resposta do plugin à solicitação de registro.

direction	plugin → agent
response	n/a

Parâmetros específicos para respostas de registro:

Nome	Tipo	Comentários
name	string	Nome do plugin.
metrics	array of strings (optional)	As métricas com descrições são usadas no plugin. Retorna RegisterMetrics(). Ausente se retornar erro.
interfaces	uint32 (optional)	A máscara de bits das interfaces suportadas pelo plugin. Ausente se retornar erro.
error	string (optional)	Uma mensagem de erro retorna caso o plugin não possa ser iniciado. Ausente se as métricas retornarem.

Exemplos:

```
{"id":2,"type":3,"metrics":["external.test", "External exporter Test."], "interfaces": 4}
```

or

```
{"id":2,"type":3,"error":"error message"}
```

Start da request

Um request para executar a função Start da interface Runner.

direction	agent → plugin
response	no

O request (solicitação) não possui parâmetros específicos, contém apenas parâmetros **common data**.

Exemplo:

```
{"id":3,"type":4}
```

Encerramento da request

Um request enviado pelo agente para desativar o plugin.

direction	agent → plugin
-----------	----------------

response	no
----------	----

Essa solicitação (request) não contém parâmetros específicos, contém apenas parâmetros **common data**.

Exemplo:

```
{"id":3,"type":5}
```

Request exportação

Um request para executar a função Exportação da interface Exporter.

direction	agent → plugin
response	no

Parâmetros específicos para request de exportação:

Nome	Tipo	Comentários
key	string	plugin key.
parameters	array of strings (optional)	Parâmetros para Função Exportação.

Exemplo:

```
{"id":4,"type":6,"key":"test.key","parameters":["foo","bar"]}
```

Exportação da response

Resposta da função Export da interface Exporter. || |---|---| |direction|plugin → agent| |response|n/a|

Parâmetros específicos para respostas de exportação:

Nome	Tipo	Comentários
value	string (optional)	Valor de resposta da função Export. Ausente se retornar erro.
error	string (optional)	Uma mensagem de erro irá aparecer caso a função Export não tiver sido executada com sucesso. Ausente se o valor retornar.

Exemplos:

```
{"id":5,"type":7,"value":"response"}
```

ou

```
{"id":5,"type":7,"error":"error message"}
```

Configuração da request

Um request para executar a função Configure da interface Configurator.

direction	agent → plugin
response	n/a

Parâmetros específicos para requests Configure:

Nome	Tipo	Comentários
global_options	JSON object	Objeto JSON contém opções de configuração globais do agente.
private_options	JSON object (optional)	Objeto JSON contém opções de configuração privada do plugin, se fornecida.

Exemplo:


```
{"id":6,"type":8,"global_options":{"..."},"private_options":{"..."}
```

Validar request

Um request para executar a função Validate na interface Configurator.

direction	agent → plugin
response	yes

Parâmetros específicos para requests Validate:

Nome	Tipo	Comentários
private_options	JSON object (optional)	Objeto JSON contém opções de configuração privada para o plugin, se fornecidas.

Exemplo:

```
{"id":7,"type":9,"private_options":{"..."}
```

Validar response

Resposta da função Validate da interface Configurator.

direction	plugin → agent
response	n/a

Parâmetros específicos para respostas Validate:

Nome	Tipo	Comentários
error	string (optional)	Uma mensagem de erro retornará caso a função Validate não for executada com sucesso. Ausente se retornar com sucesso.

Exemplo:

```
{"id":8,"type":10}
```

ou

```
{"id":8,"type":10,"error":"error message"}
```

5 Protocolo de exportação em tempo real

Esta seção apresenta detalhes do **tempo real export** em um formato JSON delimitado por nova linha para:

- **eventos de gatilho**
- **valores do item**
- **tendências**

Todos os arquivos têm uma extensão .ndjson. Cada linha do arquivo de exportação é um Objeto JSON.

Eventos de Triggers

As seguintes informações são exportadas para um evento de problema:

Campo	Tipo	Descrição
clock	number	Número de segundos desde Epoch até o momento em que o problema foi detectado (parte inteira).
ns	number	Número de nanossegundos a serem adicionados ao clock para obter um tempo preciso de detecção do problema.

Campo	Tipo		Descrição
valor		número	1 (sempre).
eventid		number	ID do evento do problema.
nome		string	Nome do evento do problema.
severity		number	Gravidade do evento do problema (0 - Não classificado, 1 - Informação, 2 - Aviso, 3 - Médio, 4 - Alto, 5 - Desastre).
hosts		array	Lista de hosts envolvidos na expressão de trigger; deve haver pelo menos um elemento no array.
	-	objeto	
	host	string	Nome do host.
	nome	string	Nome do host visível.
groups		array	lista de grupos de hosts de todos os hosts envolvidos na expressão de trigger; deve haver pelo menos um elemento no array.
	-	string	Nome do grupo de hosts.
tags		array	Lista de tags de problema (pode estar vazia).
	-	objeto	
	tag	string	Nome da tag.
	valor	string	Valor da tag (pode estar vazio).

As seguintes informações são exportadas para um evento de recuperação:

Campo	Tipo		Descrição
clock		number	Número de segundos desde a Epoch até o momento em que o problema foi resolvido (parte inteira).
ns		number	Número de nanossegundos a serem adicionados ao clock para obter um tempo preciso de resolução do problema.
valor		número	0 (sempre).
eventid		number	ID do evento de recuperação.
p_eventid		number	ID do evento do problema.

Exemplos

Problema:

```
{"clock":1519304285,"ns":123456789,"value":1,"name":"0 agente Zabbix está inacessível no Host B ou os pesq
```

Recuperação:

```
{"clock":1519304345,"ns":987654321,"value":0,"eventid":43,"p_eventid":42}
```

Problema (geração de eventos de problemas múltiplos):

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"0 agente Zabbix está inacessível no Host
```

```
{"clock":1519304286,"ns":123456789,"value":1,"eventid":43,"name":"0 agente Zabbix está inacessível no Host
```

Recuperação:

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":43}
```

```
{"clock":1519304346,"ns":987654321,"value":0,"eventid":44,"p_eventid":42}
```

Valores dos itens

As seguintes informações são exportadas para um valor de item coletado:

Campo	Tipo		Descrição
host		object	Nome do host do item host.
	host	string	Nome do host.
	name	string	Nome do host visível.
grupos		array	Lista de grupos de hosts do host do item; deve haver pelo menos um elemento no array.

Campo		Tipo	Descrição
	-	string	Nome do grupo de hosts.
itemid		número	ID do item.
nome		string	Nome do item visível.
clock		number	Número de segundos desde Epoch até o momento em que o valor foi coletado (parte inteira).
ns		number	Número de nanossegundos a serem adicionados ao clock para obter um tempo de coleta de valor preciso.
timestamp (Log apenas)		number	0 se não estiver disponível.
source (Log apenas)		string	String vazia se não estiver disponível.
gravidade (Log apenas)		número	0 se não estiver disponível.
eventid (Log apenas)		number	0 se não estiver disponível.
valor		número (para itens numéricos) ou string (para itens de texto)	Valor do item coletado.
tipo		número	Tipo de valor coletado: 0 - numérico flutuante, 1 - caractere, 2 - log, 3 - numérico sem sinal, 4 - texto

Exemplos

Valor numérico (sem sinal):

```
{"host":{"host":"Host B","name":"Host B visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":3,"na
```

Valor numérico (flutuante):

```
{"host":{"host":"Host B","name":"Host B visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":4,"na
```

Caractere, valor do texto:

```
{"host":{"host":"Host B","name":"Host B visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":2,"na
```

Valor do registro:

```
{"host":{"host":"Host A","name":"Host A visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":1,"na
```

Trends

As seguintes informações são exportadas para um valor de trends calculado:

Campo		Tipo	Descrição
host		object	Nome do host do item host.
	host	string	Nome do host.
	name	string	Nome do host visível.
grupos		array	Lista de grupos de hosts do host do item; deve haver pelo menos um elemento no array.
	-	string	Nome do grupo de hosts.
itemid		número	ID do item.
nome		string	Nome do item visível.
clock		number	Número de segundos desde Epoch até o momento em que o valor foi coletado (parte inteira).
count		number	Número de valores coletados para uma determinada hora.
min		number	Valor mínimo do item para uma determinada hora.
avg		number	Valor médio do item para uma determinada hora.
max		number	Valor máximo do item para uma determinada hora.
tipo		número	Tipo de valor: 0 - numérico flutuante, 3 - numérico sem sinal

Exemplos

```
{"host":{"host":"Host B","name":"Host B visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":3,"na
```

```
{"host":{"host":"Host B","name":"Host B visível"},"groups":["Grupo X","Grupo Y","Grupo Z"],"itemid":4,"na
```

- Os itens marcados com "X" são suportados, os marcados com "-" não são suportados.

- Note:**
Somente Windows Os itens do agente Zabbix não estão incluídos nesta tabela.

[illegible]

Attention:

O total de `vm.memory.size[used]` e `vm.memory.size[available]` não necessariamente será igual ao total. Por exemplo, em FreeBSD: memória 'active, inactive, wired, cached' são consideradas como utilizadas, pois elas podem armazenar informações úteis.

Da mesma forma as memórias 'inactive, cached, free' podem ser consideradas como disponíveis, pois estes tipos de memórias podem ser atribuídas imediatamente a processos que requeiram mais memória.

Então a memória inativa é considerada como utilizada e como disponível ao mesmo tempo. Por causa disso o item `vm.memory.size[used]` é desenvolvido para propósitos informativos apenas, enquanto o item `vm.memory.size[available]` foi desenvolvido para ser utilizado em triggers.

Note:

Consulte a seção "[Veja também](#)" ao final desta página para mais detalhes sobre o cálculo de memória em diferentes SOs.

Visão geral

Esta seção fornece alguns detalhes de parâmetros para o `vm.memory.size[<mode>]` item do agente.

Parâmetros

Os seguintes parâmetros estão disponíveis para este item:

- **ativo** - memória atualmente em uso ou usada muito recentemente, e assim esta na memória
- **anon** - memória não associada a um arquivo (não pode ser lido de isto)
- **disponível** - memória disponível, calculada de forma diferente dependendo na plataforma (veja a tabela abaixo)
- **buffers** - cache para itens como metadados do sistema de arquivos
- **cache** - cache para várias coisas
- **exec** - código executável, normalmente de um arquivo (programa)
- **arquivo** - cache para conteúdo de arquivos acessados recentemente
- **grátis** - memória que está prontamente disponível para qualquer entidade solicitante memória
- **inativo** - memória marcada como não usada
- **pavailable** - memória 'disponível' como porcentagem do 'total' (calculado como $\text{disponível} / \text{total} * 100$)
- **fixado** - igual a 'com fio'
- **pusado** - memória 'usada' como porcentagem do 'total' (calculada como $\text{usado} / \text{total} * 100$)
- **compartilhada** - memória que pode ser acessada simultaneamente por vários processos
- **slab** - quantidade total de memória usada pelo kernel para armazenar dados em cache estruturas para uso próprio
- **total** - memória física total disponível
- **usado** - memória usada, calculada de forma diferente dependendo do plataforma (veja a tabela abaixo)
- **com fio** - memória marcada para ficar sempre na RAM. Nunca é movido para o disco.

Alguns desses parâmetros são específicos da plataforma e pode não estar disponível em sua plataforma. Consulte [Itens suportados por plataforma](#) para detalhes.

Cálculo específico da plataforma de **disponível** e **usado**:

Plataforma	"disponível"	"usado"
AIX	livre + em cache	memória real em uso
FreeBSD	inativo + em cache + livre	ativo + com fio + em cache
HP UX	grátis	total - grátis
Linux<3.14	livre + buffers + cache	total - livre
Linux 3.14+ (também retroportado para 3.10 no RHEL 7)	/proc/meminfo, veja "MemAvailable" no kernel Linux documentação para detalhes. Observe que livre + buffers + cache não é mais igual a 'disponível' porque nem todo o cache da página pode ser liberado e marca d'água baixa sendo usada no cálculo.	total - livre
NetBSD	inativo + execpages + arquivo + livre	total - livre
OpenBSD	inativo + livre + em cache	ativo + com fio
OSX	inativo + livre	ativo + com fio
Solaris	grátis	total - grátis
Win32	grátis	total - grátis

::: não importante A soma de `vm.memory.size[used]` e `vm.memory.size[available]` não é necessariamente igual ao total. Para instância, no FreeBSD:

- * Memórias ativas, inativas, com fio e em cache são consideradas usadas, porque eles armazenam algumas informações úteis.
- * Ao mesmo tempo, memórias inativas, armazenadas em cache e livres são consideradas disponíveis, porque esses tipos de memórias podem ser dados instantaneamente processos que solicitam mais memória.

Assim, a memória inativa é usada e disponível simultaneamente. Por causa de isso, o item `vm.memory.size[used]` é projetado para fins informativos apenas para fins, enquanto `vm.memory.size[available]` foi projetado para ser usado em gatilhos.
...

Veja também

1. [Detalhes adicionais sobre cálculo de memória em diferentes SO](#)

2 Passive and active agent checks

Visão geral

Esta seção fornece detalhes sobre verificações passivas e ativas realizadas por **agente Zabbix**.

O Zabbix usa um protocolo de comunicação baseado em JSON para se comunicar com Agente Zabbix.

Verificações passivas

Uma verificação passiva é uma solicitação de dados simples. Servidor ou proxy Zabbix pergunta para alguns dados (por exemplo, carga da CPU) e o agente Zabbix envia de volta o resultado para o servidor.

Solicitação do servidor

Para definição de cabeçalho e comprimento de dados, consulte [protocolo detalhes](#).

<chave de item>

Resposta do agente

<DADOS> [\0<ERRO>]

Acima, a parte entre colchetes é opcional e só é enviada para não itens suportados.

Por exemplo, para itens suportados:

1. O servidor abre uma conexão TCP
2. O servidor envia **<HEADER><DATALEN>agent.ping**
3. O agente lê a solicitação e responde com **<HEADER><DATALEN>1**
4. O servidor processa os dados para obter o valor, '1' no nosso caso
5. A conexão TCP está fechada

Para itens não suportados:

1. O servidor abre uma conexão TCP
2. O servidor envia **<HEADER><DATALEN>vfs.fs.size[/nono]**
3. O agente lê a solicitação e responde com **<HEADER><DATALEN>ZBX_NOTSUPPORTED\0Não é possível obter informações do sistema de arquivos: [2] Arquivo ou diretório inexistente**
4. O servidor processa os dados, altera o estado do item para não suportado com o mensagem de erro especificada
5. A conexão TCP está fechada

Verificações ativas

As verificações ativas requerem um processamento mais complexo. O agente deve primeiro recuperar do(s) servidor(es) uma lista de itens para processamento independente.

Os servidores para obter as verificações ativas estão listados no Parâmetro 'ServerActive' do agente [configuração arquivo](#). A frequência de perguntas para essas verificações é definido pelo parâmetro 'RefreshActiveChecks' no mesmo arquivo de configuração. No entanto, se a atualização das verificações ativas falhar, é tentado novamente após 60 segundos codificados.

O agente então envia periodicamente os novos valores para o(s) servidor(es).

Se um agente estiver atrás do firewall, você pode considerar usando apenas verificações ativas porque nesse caso você não precisaria modificar o firewall para permitir conexões de entrada iniciais.

Obtendo a lista de itens

Solicitação do agente

```
{
  "request": "verificações ativas",
  "host": "<nome do host>"
}
```

Resposta do servidor

```
{
  "resposta": "sucesso",
  "dados": [
    {
      "key": "log[/home/zabbix/logs/zabbix_agentd.log]",
      "atraso": 30,
      "lastlogsize": 0,
      "mtime": 0
    },
    {
      "key": "agente.versão",
      "atraso": 600,
      "lastlogsize": 0,
      "mtime": 0
    },
    {
      "key": "vfs.fs.size[/nono]",
      "atraso": 600,
      "lastlogsize": 0,
      "mtime": 0
    }
  ]
}
```

O servidor deve responder com sucesso. Para cada item devolvido, todas as propriedades **key**, **delay**, **lastlogsize** e **mtime** devem existir, independentemente de o item ser um item de log ou não.

Por exemplo:

1. O agente abre uma conexão TCP
2. O agente solicita a lista de cheques
3. O servidor responde com uma lista de itens (chave do item, atraso)
4. O agente analisa a resposta
5. A conexão TCP está fechada
6. O agente inicia a coleta periódica de dados

::: não importante Observe que os dados de configuração (sensíveis) podem tornar-se disponível para as partes que têm acesso ao servidor Zabbix trapper porta ao usar uma verificação ativa. Isso é possível porque qualquer pessoa pode fingir ser um agente ativo e solicitar dados de configuração do item; a autenticação não ocorre a menos que você use **criptografia** opções. :::

Enviando dados coletados

Agente envia

```
{
  "request": "dados do agente",
  "sessão": "12345678901234567890123456789012",
  "dados": [
    {
      "host": "<nome do host>",
      "key": "agente.versão",
      "valor": "2.4.0",
      "id": 1,
      "relógio": 1400675595,
      "ns": 76808644
    },
    {
      "host": "<nome do host>",
      "key": "log[/home/zabbix/logs/zabbix_agentd.log]",
      "lastlogsize": 112,

```



```

        "value": " 19845:20140621:141708.521 Iniciando o Zabbix Agent [<hostname>]. Zabbix 2.4.0 (revis
        "id": 2,
        "relógio": 1400675595,
        "ns": 77053975
    },
    {
        "host": "<nome do host>",
        "key": "vfs.fs.size[/nono]",
        "estado": 1,
        "value": "Não é possível obter informações do sistema de arquivos: [2] Arquivo ou diretório ine
        "id": 3,
        "relógio": 1400675595,
        "ns": 78154128
    }
],
"relógio": 1400675595,
"ns": 78211329
}

```

Um ID virtual é atribuído a cada valor. O ID do valor é um simples ascendente contador, único dentro de uma sessão de dados (identificado pela sessão símbolo). Este ID é usado para descartar valores duplicados que podem ser enviados em ambientes de baixa conectividade.

Resposta do servidor

```

{
  "resposta": "sucesso",
  "info": "processado: 3; falhou: 0; total: 3; segundos gastos: 0,003534"
}

```

::: não importante Se o envio de alguns valores falhar no servidor (por exemplo, porque o host ou item foi desabilitado ou excluído), o agente não tente enviar novamente esses valores. :::

Por exemplo:

1. O agente abre uma conexão TCP
2. O agente envia uma lista de valores
3. O servidor processa os dados e envia o status de volta
4. A conexão TCP está fechada

Observe como no exemplo acima o status não suportado para `vfs.fs.size[/nono]` é indicado pelo valor "state" de 1 e o mensagem de erro na propriedade "valor".

::: não importante A mensagem de erro será cortada para 2048 símbolos em lado do servidor. :::

Protocolo XML antigo

Note:

O Zabbix irá tratar até 16 MB de dados XML codificados em Base64, mas um único valor decodificado não poderá ser superior a 64 KB ou será truncado para 64 KB durante a decodificação.

Veja também

1. [Mais detalhes sobre o protocolo do Zabbix Agent \(inglês\)](#)

4 Codificação dos valores retornados

O Zabbix Server espera que qualquer texto retornado esteja codificado em UTF8. Isso está relacionado a qualquer tipo de verificação: zabbix agent, ssh, telnet, etc.

Diferentes sistemas, dispositivos e verificações podem retornar caracteres não-ASCII no valor. Para estes casos, sempre que possível, as chaves dos itens no Zabbix possuem um parâmetro adicional: **<encoding>**. Este parâmetro é opcional mas pode definir que os resultado não estará com a codificação UTF8 e poderá conter caracteres não-ASCII. De outra forma, o resultado se torna imprevisível.

A seguir descrevemos o comportamento de diferentes bancos de dados nestes casos.

MySQL

Se um valor contém caracteres não-ASCII e não está codificado com UTF8 - este caractere e o restante serão descartados quando o BD armazenar o valor. Não serão geradas mensagens de alerta no zabbix_server.log.

Relevante pelo menos para a versão 5.1.61 do MySQL.

PostgreSQL

Se um valor contém caracteres não-ASCII e não está codificado com UTF8 - isso fará com que a consulta SQL falhe (PGRES_FATAL_ERROR:ERROR invalid byte sequence for encoding) e o dado não será armazenado. Uma mensagem de alerta apropriada será registrada no zabbix_server.log.

Relevante pelo menos para a versão 9.1.3 do PostgreSQL

4 Itens de Captura

Visão geral

O servidor Zabbix usa um protocolo de comunicação baseado em JSON para receber dados do remetente Zabbix com a ajuda de **trapper item**.

As mensagens de solicitação e resposta devem começar com **cabeçalho e dados comprimento**.

Solicitação do remetente do Zabbix

```
{
  "request": "dados do remetente",
  "dados": [
    {
      "host": "<nome do host>",
      "chave": "armadilha",
      "valor": "valor de teste"
    }
  ]
}
```

Resposta do servidor Zabbix

```
{
  "resposta": "sucesso",
  "info": "processado: 1; falhou: 0; total: 1; segundos gastos: 0,060753"
}
```

Solicitação do remetente do Zabbix com timestamp

Alternativamente, o remetente do Zabbix pode enviar uma solicitação com um carimbo de data e hora e nanossegundos.

```
{
  "request": "dados do remetente",
  "dados": [
    {
      "host": "<nome do host>",
      "chave": "armadilha",
      "valor": "valor de teste",
      "relógio": 1516710794,
      "ns": 592397170
    },
    {
      "host": "<nome do host>",
      "chave": "armadilha",
      "valor": "valor de teste",
      "relógio": 1516710795,
      "ns": 192399456
    }
  ],
  "relógio": 1516712029,
  "ns": 873386094
}
```

Resposta do servidor Zabbix

```
{
  "resposta": "sucesso",
  "info": "processado: 2; falhou: 0; total: 2; segundos gastos: 0,060904"
}
```

5 Nível mínimo de permissão para itens de agente do Windows

Visão geral

Ao monitorar sistemas usando um agente, uma boa prática é obter métricas do host no qual o agente está instalado. Para usar o princípio do privilégio mínimo, é necessário determinar quais métricas são obtidos do agente.

A tabela neste documento permite que você selecione os direitos mínimos para garantia de operação correta do agente Zabbix.

Se um usuário diferente for selecionado para o agente trabalhar, em vez de 'LocalSystem', então para a operação do agente como um serviço do Windows, o novo usuário deve ter os direitos "Logon como um serviço" de "Local Política → Cessão de Direitos do Usuário" e o direito de criar, escrever e excluir o arquivo de log do agente Zabbix. Um usuário do Active Directory deve ser adicionado ao Grupo Usuários do Monitor de Desempenho.

Note:

Ao trabalhar com os direitos de um agente com base na grupo "mínimo tecnicamente aceitável", provisão prévia de direitos de objetos para monitoramento é necessário.

Itens de agente comuns suportados no Windows

Chave do item	Grupo de usuários	
	Recomendado	Mínimo tecnicamente aceitável (funcionalidade limitada)
agent.hostname	Convidados	Convidados
agente.ping	Convidados	Convidados
agente.variante	Convidados	Convidados
agente.versão	Convidados	Convidados
log	Administradores	Convidados
log.count	Administradores	Convidados
logrt	Administradores	Convidados
logrt.count	Administradores	Convidados
net.dns	Convidados	Convidados
net.dns.record	Convidados	Convidados
net.if.discovery	Convidados	Convidados
net.if.in	Convidados	Convidados
net.if.out	Convidados	Convidados
net.if.total	Convidados	Convidados
net.tcp.listen	Convidados	Convidados
net.tcp.port	Convidados	Convidados
net.tcp.service	Convidados	Convidados
net.tcp.service.perf	Convidados	Convidados
net.udp.service	Convidados	Convidados
net.udp.service.perf	Convidados	Convidados
proc.num	Administradores	Convidados

Chave do item	Grupo de usuários	
system.cpu.discovery	Usuários do Monitor de Desempenho	Usuários do Monitor de Desempenho
system.cpu.load	Usuários do Monitor de Desempenho	Usuários do Monitor de Desempenho
system.cpu.num	Convidados	Convidados
system.cpu.util	Usuários do Monitor de Desempenho	Usuários do Monitor de Desempenho
system.hostname	Convidados	Convidados
system.localtime	Convidados	Convidados
system.run	Administradores	Convidados
system.sw.arch	Convidados	Convidados
system.swap.size	Convidados	Convidados
system.uname	Convidados	Convidados
system.uptime	Usuários do Monitor de Desempenho	Usuários do Monitor de Desempenho
vfs.dir.count	Administradores	Convidados
vfs.dir.get	Administradores	Convidados
vfs.dir.size	Administradores	Convidados
vfs.file.cksum	Administradores	Convidados
vfs.file.contents	Administradores	Convidados
vfs.file.exists	Administradores	Convidados
vfs.file.md5sum	Administradores	Convidados
vfs.file.regexp	Administradores	Convidados
vfs.file.regmatch	Administradores	Convidados
vfs.file.size	Administradores	Convidados
vfs.file.time	Administradores	Convidados
vfs.fs.discovery	Administradores	Convidados
vfs.fs.size	Administradores	Convidados
vm.memory.size	Convidados	Convidados
web.page.get	Convidados	Convidados
web.page.perf	Convidados	Convidados
web.page.regexp	Convidados	Convidados
zabbix.stats	Convidados	Convidados

Teclas de itens específicas do Windows

Chave do item	Grupo de usuários	
	Recomendado	Mínimo técnico-mente aceitável (funcionalidade limitada)
eventlog	Leitores de log de eventos	Convidados
net.if.list	Convidados	Convidados
perf_counter	Usuários do Monitor de Desempenho	Usuários do Monitor de Desempenho
proc_info	Administradores	Convidados
service.discovery	Convidados	Convidados
service.info	Convidados	Convidados
serviços	Hóspedes	Hóspedes
wmi.get	Administradores	Convidados
vm.vmemory.size	Convidados	Convidados

5 Suporte a grandes arquivos

O suporte a grandes arquivos, muitas vezes abreviado como LFS, é o termo aplicado para a habilidade de trabalhar com arquivos maiores que 2 GB em sistemas operacionais de 32-bit. Suportado desde o Zabbix 2.0. Esta modificação afeta pelo menos: **monitoração de arquivos de log** e todos os **itens vfs.file.***. O suporte a grandes arquivos depende das capacidades do sistema no momento da compilação do Zabbix, mas é completamente desativado no Solaris 32-bit devido à incompatibilidade com 'procfs e swapctl'.

8 Observações sobre o parâmetro memtype em itens proc.mem

Visão geral

O parâmetro **memtype** é suportado no Linux, AIX, FreeBSD, e Solaris.

Os três tipos mais comuns do 'memtype' são suportados em todas as plataformas: **pmem**, **rss** e **vsiz**. Adicionalmente, tipos específicos de cada plataforma serão nelas suportados.

AIX

Consulte os valores suportados para o parâmetro 'memtype' no AIX na tabela.

[Valor suportado]	[Descrição]	[Fonte na estrutura procentry64]	[Tenta ser compatível com]	[-----]	[-----]	[-----]	[-----]
vsiz	Tamanho da memória virtual	pi_size	<	pmem	Porcentagem de memória real	pi_prm	ps -o pmem
trss	Tamanho do conjunto residente	pi_trss + pi_drss	ps -o rssize	tamanho	Tamanho do processo (código + dados)	pi_dvm	ps gvw" coluna SIZE
dsiz	Tamanho dos dados	pi_dsiz	<	tsiz	Tamanho do texto (código)	pi_tsiz	coluna "ps gvw" TSIZ
sdsiz	Tamanho dos dados da biblioteca compartilhada	pi_sdsiz	<	drss	Tamanho do conjunto residente de dados	pi_drss	<
trss	Tamanho do conjunto residente de texto	pi_trss	<				

Notes for AIX:

1. When choosing parameters for proc.mem[] item key on AIX, try to specify narrow process selection criteria. Otherwise there is a risk of getting unwanted processes counted into proc.mem[] result.

Example:

```
\$ zabbix_agentd -t proc.mem[,,,NonExistingProcess,rss]
proc.mem[,,,NonExistingProcess,rss] [u|2879488]
```

This example shows how specifying only command line (regular expression to match) parameter results in Zabbix agent self-accounting - probably not what you want.

2. Do not use "ps -ef" to browse processes - it shows only non-kernel processes. Use "ps -Af" to see all processes which will be seen by Zabbix agent.
3. Let's go through example of 'topasrec' how Zabbix agent proc.mem[] selects processes.

```
\$ ps -Af | grep topasrec
root 10747984      1   0   Mar 16      -   0:00 /usr/bin/topasrec  -L -s 300 -R 1 -r 6 -o /var/perf daily
```

proc.mem[] has arguments:

```
proc.mem[<name>,<user>,<mode>,<cmdline>,<memtype>]
```

The 1st criterion is a process name (argument <name>). In our example Zabbix agent will see it as 'topasrec'. In order to match, you need to either specify 'topasrec' or to leave it empty. The 2nd criterion is a user name (argument <user>). To match, you need to either specify 'root' or to leave it empty. The 3rd criterion used in process selection is an argument <cmdline>. Zabbix agent will see its value as '/usr/bin/topasrec -L -s 300 -R 1 -r 6 -o /var/perf/daily/ -ypersistent=1 -O type=bin -ystart_time=04:08:54,Mar16,2023'. To match, you need to either specify a regular expression which matches this string or to leave it empty.

Arguments <mode> and <memtype> are applied after using the three criteria mentioned above.

FreeBSD

Os valores a seguir são suportados para o parâmetro 'memtype' nesta plataforma.

Valor suportado	Descrição	Fonte em estrutura kinfo_proc	Tentativas para compatibilizar
vsize	Tamanho da memória virtual	kp_eproc.e_vm.vm_map.size or ki_size	ps -o vsz
pmem	Percentual da memória real	calculado a partir do rss	ps -o pmem
rss	Tamanho do conjunto residente	kp_eproc.e_vm.vm_rssize or ki_rssize	ps -o rss
size ((- valor padrão))	Tamanho do processo (código + dados + pilha)	tsize + dsize + ssize	
tsize	Tamanho do texto de código	kp_eproc.e_vm.vm_tsize or ki_tsize	ps -o tsiz
dsize	Tamanho dos dados	kp_eproc.e_vm.vm_dsize or ki_dsize	ps -o dsiz
ssize	Tamanho da pilha	kp_eproc.e_vm.vm_ssize or ki_ssize	ps -o ssiz

Linux

Consulte os valores suportados para o parâmetro 'memtype' no Linux na tabela.

[Valor suportado|Descrição|Fonte no arquivo /proc/<pid>/status| |-----|-----|-----| |vsize¹|Tamanho da memória virtual|VmSize| |pmem|Porcentagem de memória real|(VmRSS/total_memory) * 100| |rss|Tamanho do conjunto residente|VmRSS| |dados|Tamanho do segmento de dados|VmData| |exe|Tamanho do segmento de código|VmExe| |hwm|Tamanho do conjunto residente de pico|VmHWM| |lck|Tamanho da memória bloqueada|VmLck| |lib|Tamanho das bibliotecas compartilhadas|VmLib| |pico|Tamanho da memória virtual de pico|VmPeak| |pin|Tamanho das páginas fixadas|VmPin| |pte|Tamanho das entradas da tabela de páginas|VmPTE| |size|Tamanho do código do processo + dados + segmentos de pilha|VmExe + VmData + VmStk| |stk|Tamanho do segmento de pilha|VmStk| |swap|Tamanho do espaço de troca usado|VmSwap|

Notas para Linux:

1. Nem todos os valores 'memtype' são suportados por kernels Linux mais antigos. Por exemplo, kernels Linux 2.4 não suportam hwm, pin, peak, valores pte e swap.

2. Percebemos que o automonitoramento do agente Zabbix ativo processo de verificação com `proc.mem[...,...,...,data]` mostra um valor que é 4 kB maior do que o relatado pela linha `VmData` no agente arquivo `/proc/<pid>/status`. No momento da auto-medição, o segmento de dados do agente aumenta em 4 kB e então retorna ao tamanho anterior.

Solaris

Os valores a seguir são suportados para o parâmetro 'memtype' nesta plataforma.

Valor suportado	Descrição	Fonte em estrutura psinfo	Tentativas para compatibilizar
vsz ((- valor padrão))	Tamanho da imagem do processo	pr_size	ps -o vsz
pmem	Percentual da memória real	pr_pctmem	ps -o pmem
rss	Tamanho do conjunto residente Que pode ser subestimado, conforme descrição do rss em "man ps".	pr_rssize	ps -o rss

Notas de rodapé

¹ Valor padrão.

8 Sensor

Cada chip sensor obtém seu próprio diretório na árvore `sysfs /sys/devices`. Para encontrar todos os chips do sensor, é mais fácil seguir os links simbólicos do dispositivo de `/sys/class/hwmon/hwmon*`, onde * é um número real (0,1,2,...).

As leituras do sensor estão localizadas em `/sys/class/hwmon/hwmon*/` diretório para dispositivos virtuais ou em `/sys/class/hwmon/hwmon*/dev` diretório para dispositivos não virtuais. Um arquivo, chamado nome, localizado dentro `hwmon*` ou `hwmon*/device` diretórios contém o nome do chip, que corresponde ao nome do driver do kernel usado pelo sensor `lasca`.

Existe apenas um valor de leitura do sensor por arquivo. O esquema comum de nomear os arquivos que contêm leituras do sensor dentro de qualquer um dos diretórios mencionados acima é: `<tipo><número>_<item>`, onde

- **tipo** - para chips de sensor é "in" (tensão), "temp" (temperatura), "fan" (ventilador), etc.,
- **item** - "input" (valor medido), "max" (limite alto), "min" (limite baixo), etc.,
- **número** - sempre usado para elementos que podem estar presentes mais de uma vez (geralmente começa em 1, exceto para tensões que começam em 0). Se os arquivos não se referem a um elemento específico, eles têm um simples nome sem número.

As informações sobre os sensores disponíveis no host podem ser adquiridas usando as ferramentas **sensor-detect** e **sensors** (pacote `lm-sensors`: <http://lm-sensors.org/>). **Sensors-detect** ajuda a determinar quais módulos são necessários para os sensores disponíveis. Quando os módulos são carregados, o **sensors** programa pode ser usado para mostrar as leituras de todos os sensores salgadinhos. A rotulagem das leituras dos sensores, utilizada por este programa, pode ser diferente do esquema de nomenclatura comum (`<tipo><número>_<item>`):

- se houver um arquivo chamado `<type><number>_label`, então o rótulo dentro deste arquivo será usado em vez de `<tipo><número><item>` nome;
- se não houver arquivo `<type><number>_label`, então o programa procura dentro do `/etc/sensors.conf` (pode ser também `/etc/sensors3.conf` ou diferente) para a substituição do nome.

Esta rotulagem permite ao usuário determinar que tipo de hardware é usado. Se não há arquivo `<type><number>_label` nem rótulo dentro no arquivo de configuração o tipo de hardware pode ser determinado pelo atributo `name` (`hwmon*/device/name`). Os nomes reais dos sensores, que `zabbix_agent` aceita, pode ser obtido executando o programa **sensors** com o parâmetro `-u` (**sensors -u**).

No programa **sensor** os sensores disponíveis são separados pelo barramento tipo (adaptador ISA, adaptador PCI, adaptador SPI, dispositivo virtual, ACPI interface, adaptador HID).

No Linux 2.4:

(A leitura dos sensores pode ser obtida a partir do diretório `/proc/sys/dev/sensors`)

- **device** - nome do dispositivo (se o `<mode>` for utilizado, será uma expressão regular);
- **sensor** - nome do sensor (se o `<mode>` for utilizado, será uma expressão regular);
- **mode** - valores possíveis: avg, max, min (se este parâmetro for omitido o dispositivo e o sensor serão tratados de forma textual).

Exemplo de chave: `sensor[w83781d-i2c-0-2d,temp1]`

Antes do Zabbix 1.8.4, o formato `sensor[temp1]` era utilizado.

No Linux 2.6+:

(A leitura dos sensores pode ser obtida a partir do diretório `/sys/class/hwmon`)

- **device** - nome do dispositivo (expressão não regular). O nome do dispositivo pode ser o nome atual do dispositivo (ex. `0000:00:18.3`) ou o nome obtido usando os programas de sensor (ex.. `k8temp-pci-00c3`). Cabe ao usuário escolher qual nome utilizar;
- **sensor** - nome do sensor (expressão não regular);
- **mode** - valores possíveis: avg, max, min (se este parâmetro for omitido o dispositivo e o sensor serão tratados de forma textual).

Exemplos de chave:

`sensor[k8temp-pci-00c3,temp, max]` ou `sensor[0000:00:18.3,temp1]`

`sensor[sm5c47b397-isa-0880,in, avg]` ou `sensor[sm5c47b397.2176,in1]`

Obtendo nomes de sensores

As etiquetas dos sensores, conforme impressas pelo comando `sensors`, nem sempre podem ser usado diretamente porque a nomenclatura dos rótulos pode ser diferente para cada fornecedor do chip sensor. Por exemplo, a saída de sensores pode conter o seguintes linhas:

```
$ sensors
in0: +2,24 V (min = +0,00 V, máx = +3,32 V)
Vcore: +1,15 V (mín. = +0,00 V, máx. = +2,99 V)
+3,3 V: +3,30 V (mín. = +2,97 V, máx. = +3,63 V)
+12 V: +13,00 V (mín. = +0,00 V, máx. = +15,94 V)
M/B Temp: +30,0°C (baixo = -127,0°C, alto = +127,0°C)
```

Destes, apenas um rótulo pode ser usado diretamente:

```
$ zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in0]
2,240000
```

Tentar usar outros rótulos (como `Vcore` ou `+12V`) não funcionará.

```
$ zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,Vcore]
ZBX_NOTSUPPORTED
```

Para descobrir o nome real do sensor, que pode ser usado pelo Zabbix para recupere as leituras do sensor, execute `sensors -u`. Na saída, o podem ser observados:

```
$ sensors -u
...
Vcore:
  entrada1_entrada: 1,15
  in1_min: 0,00
  in1_max: 2,99
  in1_alarme: 0,00
...
+12V:
  entrada4_entrada: 13,00
  em 4_min: 0,00
  in4_max: 15,94
  in4_alarme: 0,00
...
```

Portanto, `Vcore` deve ser consultado como `in1` e `+12V` deve ser consultado como `em4`.⁷

⁷ De acordo com [especificação](#) estas são voltagens nos pinos do chip e, de um modo geral, podem precisar dimensionamento.


```
$ zabbix_get -s 127.0.0.1 -k sensor[lm85-i2c-0-2e,in1]
1,301000
```

Não apenas tensão (in), mas também corrente (curr), temperatura (temp) e as leituras de velocidade do ventilador (ventilador) podem ser recuperadas pelo Zabbix.

9 Observações sobre seleção de processos nos itens proc.mem e proc.num

Processos modificando sua linha de comando

Alguns programas usam a modificação de sua linha de comando como um método para exibir sua atividade atual. Um usuário pode ver a atividade executando ps e comandos top. Exemplos de tais programas incluem PostgreSQL, Sendmail, Zabbix.

Vamos ver um exemplo do Linux. Vamos supor que queremos monitorar um número de processos do agente Zabbix.

O comando ps mostra os processos de interesse como

```
$ ps -fu zabbix
UID PID PPID C STIME TTY TIME CMD
...
zabbix 6318 1 0 12:01 ? 00:00:00 sbin/zabbix_agentd -c /home/zabbix/ZBXNEXT-1078/zabbix_agentd.conf
zabbix 6319 6318 0 12:01 ? 00:00:01 sbin/zabbix_agentd: coletor [inativo 1 segundo]
zabbix 6320 6318 0 12:01 ? 00:00:00 sbin/zabbix_agentd: listener #1 [aguardando conexão]
zabbix 6321 6318 0 12:01 ? 00:00:00 sbin/zabbix_agentd: listener #2 [aguardando conexão]
zabbix 6322 6318 0 12:01 ? 00:00:00 sbin/zabbix_agentd: listener #3 [aguardando conexão]
zabbix 6323 6318 0 12:01 ? 00:00:00 sbin/zabbix_agentd: verificações ativas #1 [ocioso 1 segundo]
...
```

Selecionar processos por nome e usuário faz o trabalho:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd,zabbix] '
6
```

Agora vamos renomear o executável zabbix_agentd para zabbix_agentd_30 e reinicie-o.

ps agora mostra

```
$ ps -fu zabbix
UID PID PPID C STIME TTY TIME CMD
...
zabbix 6715 1 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30 -c /home/zabbix/ZBXNEXT-1078/zabbix_agentd.conf
zabbix 6716 6715 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30: coletor [inativo 1 segundo]
zabbix 6717 6715 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30: ouvinte #1 [aguardando conexão]
zabbix 6718 6715 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30: ouvinte #2 [aguardando conexão]
zabbix 6719 6715 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30: ouvinte #3 [aguardando conexão]
zabbix 6720 6715 0 12:53 ? 00:00:00 sbin/zabbix_agentd_30: verificações ativas #1 [ocioso 1 segundo]
...
```

Agora, selecionar processos por nome e usuário produz um resultado incorreto:

```
$ zabbix_get -s localhost -k 'proc.num[zabbix_agentd_30,zabbix] '
1
```

Por que uma simples renomeação de executável para um nome mais longo leva a resultado diferente?

O agente Zabbix começa verificando o nome do processo. /proc/<pid>/status arquivo é aberto e a linha Name é marcada. No nosso caso, o Nome linhas são:

```
$ grep Nome /proc/{6715,6716,6717,6718,6719,6720}/status
/proc/6715/status:Nome: zabbix_agentd_3
/proc/6716/status:Nome: zabbix_agentd_3
/proc/6717/status:Nome: zabbix_agentd_3
/proc/6718/status:Nome: zabbix_agentd_3
/proc/6719/status:Nome: zabbix_agentd_3
/proc/6720/status:Nome: zabbix_agentd_3
```

O nome do processo no arquivo status é truncado para 15 caracteres.

Um resultado semelhante pode ser visto com o comando ps:

```
$ ps -u zabbix
  PID TTY TIME CMD
...
6715? 00:00:00 zabbix_agentd_3
6716? 00:00:01 zabbix_agentd_3
6717? 00:00:00 zabbix_agentd_3
6718? 00:00:00 zabbix_agentd_3
6719? 00:00:00 zabbix_agentd_3
6720? 00:00:00 zabbix_agentd_3
...
```

Obviamente, isso não é igual ao valor do nosso parâmetro `proc.num[] name zabbix_agentd_30`. Não ter correspondido ao nome do processo de arquivo `status` o agente Zabbix transforma em arquivo `/proc/<pid>/cmdline`.

Como o agente vê o arquivo "cmdline" pode ser ilustrado executando um comando

[illegible]

Os arquivos `/proc/<pid>/cmdline` em nosso caso contêm arquivos invisíveis, não imprimíveis bytes nulos, usados para encerrar strings na linguagem C. Os bytes nulos são mostrados como `"<NUL>"` neste exemplo.

O agente Zabbix verifica "cmdline" para o processo principal e leva um zabbix_agentd_30, que corresponde ao nosso valor de parâmetro name zabbix_agentd_30. Assim, o processo principal é contado por item proc.num[zabbix_agentd_30,zabbix].

Ao verificar o próximo processo, o agente leva `zabbix_agentd_30: collector [idle 1 sec]` do arquivo `cmdline` e ele não atende ao nosso parâmetro `name zabbix_agentd_30`. Então, apenas o processo principal que não modifica sua linha de comando, é contado. De outros os processos do agente modificam sua linha de comando e são ignorados.

Este exemplo mostra que o parâmetro `name` não pode ser usado em `proc.mem[]` e `proc.num[]` para selecionar processos neste caso.

Usar o parâmetro `cmdline` com uma expressão regular adequada produz um resultado correto:

```
$ zabbix_get -s localhost -k 'proc.num[,zabbix,,zabbix_agentd_30[:]]'
6
```

Tenha cuidado ao usar os itens `proc.mem[]` e `proc.num[]` para monitoramento programas que modificam suas linhas de comando.

Antes de colocar os parâmetros `name` e `cmdline` em `proc.mem[]` e itens `proc.num[]`, você pode querer testar os parâmetros usando item `proc.num[]` e comando `ps`.

Threads do kernel Linux

Threads não podem ser selecionadas com o parâmetro `cmdline` nos itens `proc.mem[]` e `proc.num[]`

Vamos ver um exemplo das threads do kernel:

```
$ ps -ef | grep kthreadd
root      2      0  0 09:33 ?          00:00:00 [kthreadd]
```

Ela pode ser selecionada através do parâmetro `name`:

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd,root]'
1
```

Mas selecionado pelo parâmetro `cmdline` não irá funcionar:

```
$ zabbix_get -s localhost -k 'proc.num[,root,,kthreadd]'
0
```

A razão disso é que o agente utiliza a expressão regular definida no parametro `cmdline` e aplica seu conteúdo ao conteúdo do processo `/proc/<pid>/cmdline`. Mas para as threads do kernel os arquivos `/proc/<pid>/cmdline` estarão vazios. Logo, o parâmetro `cmdline` nunca irá funcionar para elas.

Contagem de threads em itens `proc.mem[]` e `proc.num[]`

threads do kernel Linux são contados pelo item `proc.num[]`, mas não relatam memória no item `proc.mem[]`. Por exemplo:

```
$ ps -ef | grep kthreadd
raiz 2 0 0 09:51 ? 00:00:00 [kthreadd]
```

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd]'
1
```

```
$ zabbix_get -s localhost -k 'proc.mem[kthreadd]'
ZBX_NOTSUPPORTED: Não é possível obter a quantidade de memória "VmSize".
```

Mas o que acontece se houver um processo de usuário com o mesmo nome de um thread do núcleo? Então poderia ficar assim:

```
$ ps -ef | grep kthreadd
raiz 2 0 0 09:51 ? 00:00:00 [kthreadd]
zabbix 9611 6133 0 17:58 pts/1 00:00:00 ./kthreadd
```

```
$ zabbix_get -s localhost -k 'proc.num[kthreadd]'
2
```

```
$ zabbix_get -s localhost -k 'proc.mem[kthreadd]'
4157440
```

`proc.num[]` contou tanto a thread do kernel quanto o processo do usuário. `proc.mem[]` reporta a memória apenas para o processo do usuário e conta a memória de thread do kernel como se fosse 0. Isso é diferente do caso acima quando `ZBX_NOTSUPPORTED` foi relatado.

Tenha cuidado ao usar os itens `proc.mem[]` e `proc.num[]` se o programa name acontece para corresponder a um dos threads.

Antes de colocar parâmetros nos itens `proc.mem[]` e `proc.num[]`, você pode querer testar os parâmetros usando o item `proc.num[]` e `ps` comando.

11 Detalhes de implementação das verificações `net.tcp.service` e `net.udp.service`

A implementação das verificações `net.tcp.service` e `net.udp.service` é detalhada nesta página para os vários serviços definidos por seus parâmetros.

Parâmetros `net.tcp.service` do item

ftp

Cria uma conexão TCP e espera os primeiros 4 caracteres do resposta seja "220 ", então envia "QUIT\r\n". A porta padrão 21 é usada se não for especificado.

http

Cria uma conexão TCP sem esperar e enviar nada. Predefinição a porta 80 é usada se não for especificada.

https

Usa (e só funciona com) libcurl, não verifica a autenticidade de o certificado, não verifica o nome do host no certificado SSL, apenas busca o cabeçalho de resposta (solicitação HEAD). A porta padrão 443 é usado se não for especificado.

imap

Cria uma conexão TCP e espera os primeiros 4 caracteres do resposta seja "* OK", então envia "a1 LOGOUT\r\n". Porta padrão 143 é usado se não for especificado.

ldap

Abre uma conexão com um servidor LDAP e executa uma pesquisa LDAP operação com filtro definido como (objectClass=*). Espera sucesso recuperação do primeiro atributo da primeira entrada. A porta padrão 389 é usado se não for especificado.

nntp

Cria uma conexão TCP e espera os 3 primeiros caracteres do resposta seja "200" ou "201", então envia "QUIT\r\n". Porta padrão 119 é usado se não for especificado.

pop

Cria uma conexão TCP e espera os 3 primeiros caracteres do resposta seja "+OK", então envia "QUIT\r\n". A porta padrão 110 é usada se não for especificado.

smtp

Cria uma conexão TCP e espera os 3 primeiros caracteres do resposta seja "220", seguido por um espaço, o final da linha ou um traço. As linhas que contêm um traço pertencem a uma resposta de várias linhas e a resposta será relido até que uma linha sem o traço seja recebida. Então envia "QUIT\r\n". A porta padrão 25 é usada se não for especificada.

ssh

Cria uma conexão TCP. Se a conexão foi estabelecida, ambos lados trocam uma string de identificação (SSH-major.minor-XXXX), onde major e minor são versões de protocolo e XXXX é uma string. Zabbix verifica se a string que corresponde à especificação foi encontrada e então envia volte a string "SSH-major.minor-zabbix_agent\r\n" ou "0\r\n" em incompatibilidade. A porta padrão 22 é usada se não for especificada.

tcp

Cria uma conexão TCP sem esperar e enviar nada. diferente as outras verificações requerem que o parâmetro port seja especificado.

telnet

Cria uma conexão TCP e espera um prompt de login (':' no final). A porta padrão 23 é usada se não for especificada.

Item net.udp.service parameters

ntp

Envia um pacote SNTP sobre UDP e valida a resposta conforme [RFC 4330, seção 5](#). O padrão é a porta 123 .

12 Definições de host inalcançável/inacessível

Visão geral

Diversos **parâmetros de configuração** definem como o Zabbix Server irá agir quando uma verificação de agente (Zabbix, SNMP, IPMI, JMX) falhar e um host passar a se tornar inacessível.

Interface inacessível

Uma interface de host é tratada como inacessível após uma falha na verificação (rede error, timeout) pelos agentes Zabbix, SNMP, IPMI ou JMX. Observe que o Zabbix as verificações ativas do agente não influenciam a disponibilidade da interface de forma alguma.

A partir desse momento, **UnreachableDelay** define com que frequência uma interface é verificado novamente usando um dos itens (incluindo regras LLD) neste situação de inacessibilidade e tais verificações já serão realizadas por pollers inacessíveis (ou pollers IPMI para verificações de IPMI). Por padrão é 15 segundos antes da próxima verificação.

No log do servidor Zabbix, a inacessibilidade é indicada por mensagens como esses:

```
O item do agente Zabbix "system.cpu.load[percpu,avg1]" no host "Novo host" falhou: primeiro erro de rede,
Item do agente Zabbix "system.cpu.load[percpu,avg15]" no host "Novo host" falhou: outro erro de rede, agua
```

Observe que o item exato que falhou é indicado e o tipo de item (agente Zabbix).

Note:

O parâmetro Timeout também afetará o quão cedo um interface é verificada novamente durante a inacessibilidade. Se o tempo limite for 20 segundos e UnreachableDelay 30 segundos, a próxima verificação será em 50 segundos após a primeira tentativa.

O parâmetro **UnreachablePeriod** define por quanto tempo a inacessibilidade período é total. Por padrão, UnreachablePeriod é de 45 segundos. UnreachablePeriod deve ser várias vezes maior que UnreachableDelay, para que uma interface seja verificada mais de uma vez antes de uma interface fica indisponível.

Alternando a interface de volta para disponível

Quando o período de inacessibilidade termina, a interface é sondada novamente, diminuindo a prioridade para o item que tornou a interface inacessível Estado. Se a interface inacessível reaparecer, o monitoramento retorna ao normal automaticamente:

retomando as verificações do agente Zabbix no host "Novo host": conexão restaurada

Note:

Uma vez que a interface se torna disponível, o host não pesquisa todos seus itens imediatamente por dois motivos:

- Pode sobrecarregar o host.
- O tempo de restauração da interface nem sempre corresponde ao item planejado horário da agenda de votação.

Assim, após a disponibilização da interface, os itens não são pesquisados imediatamente, mas eles estão sendo remarcados para a próxima votação volta.

Interface indisponível

Após o término do `UnreachablePeriod` e a interface não reaparecer, a interface é tratada como indisponível.

No log do servidor é indicado por mensagens como estas:

desabilitando temporariamente as verificações do agente Zabbix no host "Novo host": interface indisponível

E no **frontend** o ícone de disponibilidade do host vai de verde/cinza para amarelo/vermelho (o detalhes de interface inacessíveis podem ser vistos na caixa de dicas que é exibido quando um mouse é posicionado no ícone de disponibilidade do host):

ZBX Items 7 Triggers 3 Graphs Discovery rules Web scenarios 2			
Interface	Status	Error	
127.0.0.1:10050	Available		
192.0.0.1:10050	Not available	Get value from agent failed: cannot connect to [[192.0.0.1]:10050]: [4] system call	

O parâmetro **UnavailableDelay** define com que frequência uma interface é verificado durante a indisponibilidade da interface.

Por padrão, são 60 segundos (então, neste caso, "desativando temporariamente", da mensagem de log acima, significará desabilitar as verificações por um minuto).

Quando a conexão com a interface é restaurada, o monitoramento retorna ao normal automaticamente também:

habilitando as verificações do agente Zabbix no host "Novo host": a interface ficou disponível

13 Monitoramento remoto das estatísticas do Zabbix

Visão geral

É possível fazer algumas métricas internas do servidor e proxy Zabbix acessível remotamente por outra instância do Zabbix ou uma ferramenta de terceiros. Isso pode ser útil para que apoiadores/prestadores de serviços possam monitorar seus servidores/proxies Zabbix cliente remotamente ou, em organizações onde O Zabbix não é a principal ferramenta de monitoramento, que as métricas internas do Zabbix podem ser monitorado por um sistema de terceiros em uma configuração de monitoramento de guarda-chuva.

As estatísticas internas do Zabbix são expostas a um conjunto configurável de endereços listados no novo 'StatsAllowedIP' **servidor/proxy** parâmetro. As solicitações serão aceitas somente a partir desses endereços.

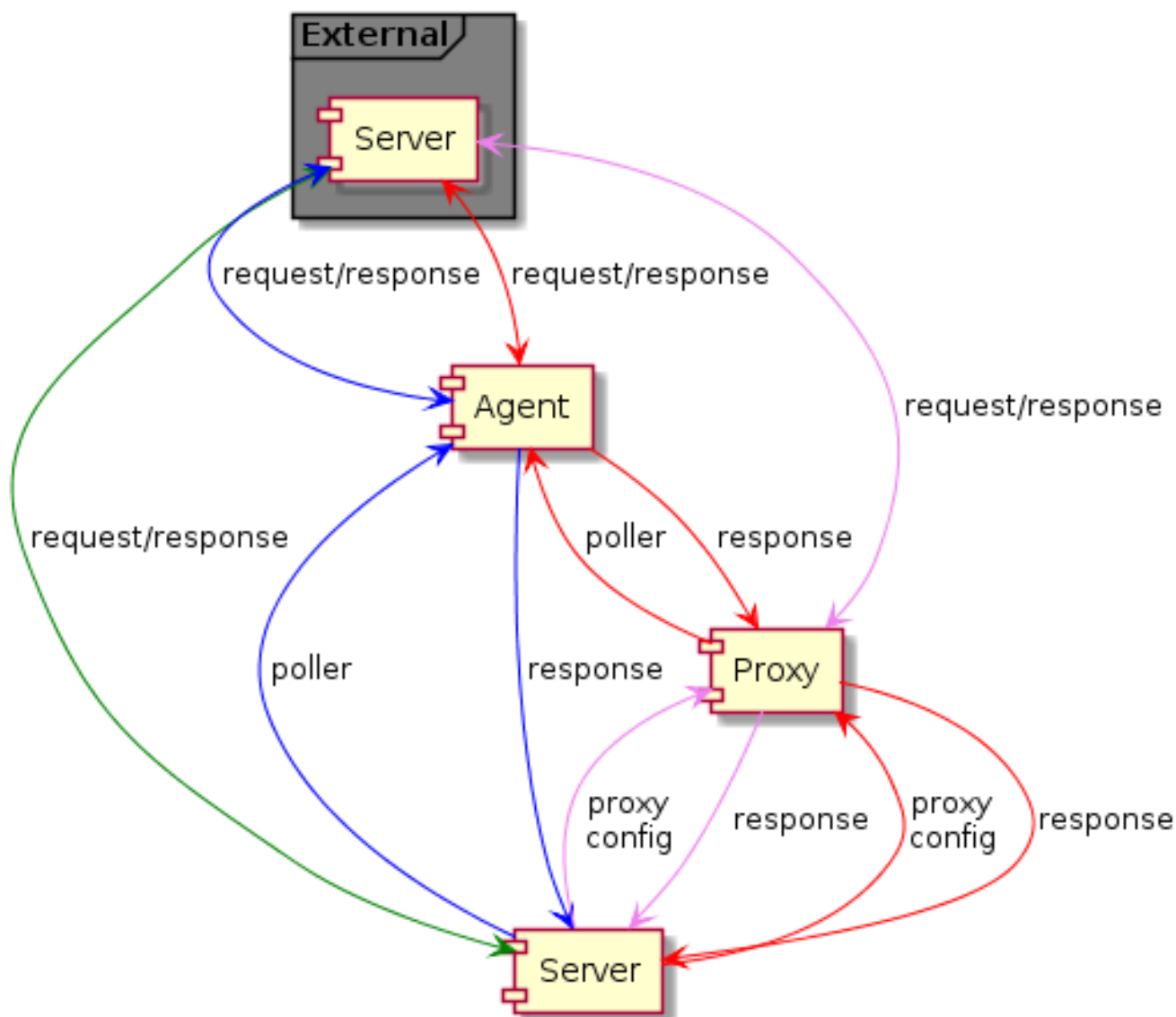
Itens

Para configurar a consulta de estatísticas internas em outra instância do Zabbix, você pode usar dois itens:

- `zabbix[stats,<ip>,<port>]` item interno - para controle remoto direto consultas do servidor/proxy Zabbix. `<ip>` e `<port>` são usados para identificar a instância de destino.
- `zabbix.stats[<ip>,<port>]` item de agente - para remoto baseado em agente consultas do servidor/proxy Zabbix. `<ip>` e `<port>` são usados para identificar a instância de destino.

Veja também: **Itens internos, itens do agente Zabbix**

O diagrama a seguir ilustra o uso de qualquer item dependendo o contexto.



- ■ - Servidor → instância externa do Zabbix (zabbix[stats,<ip>,<port>])
- ■ - Servidor → proxy → instância externa do Zabbix (zabbix[stats,<ip>,<port>])
- ■ - Servidor → agente → instância externa do Zabbix (zabbix.stats[<ip>,<port>])
- ■ - Servidor → proxy → agente → instância externa do Zabbix (zabbix.stats[<ip>,<port>])

Para certificar-se de que a instância de destino permite consultá-la pelo instância, liste o endereço da instância externa no Parâmetro 'StatsAllowedIP' na instância de destino.

Métricas expostas

Os itens de estatísticas reúnem as estatísticas em massa e retornam um JSON, que é a base para os itens dependentes obterem seus dados. A seguir **métricas internas** são retornadas por um dos dois itens:

- zabbix[boottime]
- zabbix[hosts]
- zabbix[itens]
- zabbix[items_unsupported]
- zabbix[preprocessing_queue] (somente servidor)
- zabbix[process,<type>,<mode>,<state>] (apenas o tipo de processo baseado Estatísticas)
- zabbix[rcache,<cache>,<modo>]
- zabbix[desempenho obrigatório]
- zabbix[triggers] (somente servidor)
- zabbix[tempo de atividade]
- zabbix[vcache,buffer,<mode>] (somente servidor)
- zabbix[vcache,cache,<parâmetro>]

- zabbix[versão]
- zabbix[vmware,buffer,<modo>]
- zabbix[wcache,<cache>,<mode>] (somente servidor de tipo de cache 'trends')

Templates

Os Templates estão disponíveis para **remoto monitoramento** do servidor Zabbix ou métricas internas de proxy de uma instância externa:

- Servidor Zabbix Remoto
- Proxy Zabbix remoto

Observe que, para usar um modelo para monitoramento remoto de várias instâncias externas, é necessário um host separado para cada monitoramento de instância.

Processo do Trapper

Receber solicitações de métricas internas de uma instância externa do Zabbix é tratado pelo processo trapper que valida o pedido, reúne o métricas, cria o buffer de dados JSON e envia o JSON preparado de volta, por exemplo, do servidor:

```
{
  "resposta": "sucesso",
  "dados": {
    "boottime": N,
    "tempo de atividade": N,
    "anfitriões": N,
    "itens": N,
    "items_unsupported": N,
    "preprocessing_queue": N,
    "processar": {
      "gerente de alertas": {
        "ocupado": {
          "média": N,
          "máximo": N,
          "min": N
        },
        "ocioso": {
          "média": N,
          "máximo": N,
          "min": N
        },
        "contar": N
      },
      ...
    },
    "fila": N,
    "rcache": {
      "total": N,
      "grátis": N,
      "pfree": N,
      "usado": N,
      "empurrado": N
    },
    "desempenho necessário": N,
    "gatilhos": N,
    "tempo de atividade": N,
    "vcache": {
      "amortecedor": {
        "total": N,
        "grátis": N,
        "pfree": N,
        "usado": N,
        "empurrado": N
      },
      "cache": {
        "solicitações": N,
```

```

        "acertos": N,
        "falta": N,
        "modo": N
    }
},
"vmware": {
    "total": N,
    "grátis": N,
    "pfree": N,
    "usado": N,
    "empurrado": N
},
"versão": "N",
"wcache": {
    "valores": {
        "tudo n,
        "flutuar": N,
        "uint": N,
        "str": N,
        "log": N,
        "texto": N,
        "não suportado": N
    },
    "história": {
        "pfree": N,
        "grátis": N,
        "total": N,
        "usado": N,
        "empurrado": N
    },
    "índice": {
        "pfree": N,
        "grátis": N,
        "total": N,
        "usado": N,
        "empurrado": N
    },
    "tendência": {
        "pfree": N,
        "grátis": N,
        "total": N,
        "usado": N,
        "empurrado": N
    }
}
}
}
}
}

```

Itens da fila interna

Há também outros dois itens que permitem especificamente a consulta remota estatísticas da fila interna em outra instância do Zabbix:

- `zabbix[stats,<ip>,<port>,queue,<from>,<to>]` item interno - para consultas de filas internas diretas para servidor/proxy Zabbix remoto
- `zabbix.stats[<ip>,<port>,queue,<from>,<to>]` item do agente - para consultas de filas internas baseadas em agente para servidor/proxy Zabbix remoto

Veja também: [Itens internos](#), [itens do agente Zabbix](#)

14 Configurando Kerberos com Zabbix

Visão geral

A autenticação Kerberos pode ser usada no monitoramento Web e itens HTTP em Zabbix desde a versão 4.4.0.

Esta seção descreve um exemplo de configuração do Kerberos com o Zabbix servidor para realizar o monitoramento web de `www.example.com` com usuário 'zabbix'.

Etapas

Passo 1

Instale o pacote Kerberos.

Para Debian/Ubuntu:

```
apt install krb5-user
```

Para RHEL/CentOS:

```
yum install krb5-workstation
```

Passo 2

Configure o arquivo de configuração do Kerberos (consulte a documentação do MIT para detalhes)

```
cat /etc/krb5.conf
[libdefaults]
    default_realm = EXAMPLE.COM

#### The following krb5.conf variables are only for MIT Kerberos.
    kdc_timesync = 1
    ccache_type = 4
    forwardable = true
    proxiable = true

[realms]
    EXAMPLE.COM = {
    }

[domain_realm]
    .example.com=EXAMPLE.COM
    example.com=EXAMPLE.COM
```

Etapa 3

Crie um tíquete Kerberos para o usuário zabbix. Execute o seguinte comando como usuário zabbix:

```
kinit zabbix
```

::: não importante É importante executar o comando acima como usuário zabbix. Se você executá-lo como root, a autenticação não será aceita. :::

Passo 4

Criar um cenário Web ou item de agente HTTP com autenticação Kerberos tipo.

Opcionalmente, pode ser testado com o seguinte comando curl:

```
curl -v --negotiate -u : http://example.com
```

Observe que, para um monitoramento web longo, é necessário cuidar da renovação do tíquete Kerberos. O tempo padrão de expiração é 10h.

15 parâmetros modbus.get

Visão geral

A tabela abaixo apresenta detalhes do `modbus.get[]` item parâmetros.

Parâmetros

Parâmetro	Descrição	Padrões	Exemplo
endpoint	<p>Protocolo e endereço do endpoint, definido como <code>protocol://connection_string</code></p> <p>Possíveis valores de protocolo: rtu, ascii (somente Agente 2), tcp</p> <p>Formato da string de conexão: com tcp - <code>address:port</code> com linha serial: rtu, ascii - <code>port_name:speed:params</code> onde < br> 'speed' - 1200, 9600 etc 'params' - bits de dados (5,6,7 ou 8), paridade (n,e ou o para nenhum/par/ímpar), bits de parada (1 ou 2)</p>	<p>protocolo: nenhum</p> <p>rtu/ascii protocolo: port_name: nenhum velocidade: 115200 parâmetros: 8n1</p> <p>tcp protocolo:< br>endereço: nenhum porta: 502</p>	<p>tcp://192.168.6.1:511 tcp://192.168.6.2 tcp://[::1]:511 tcp://::1 tcp://localhost:511 tcp://localhost rtu://COM1:9600:8n ascii://COM2:1200:7o2< br>rtu://ttyS0:9600 ascii://ttyS1</p>
ID do escravo	<p>Endereço Modbus do dispositivo ao qual se destina (1 a 247), consulte Guia de implementação de mensagens MODBUS (página 23)</p> <p>dispositivo tcp (não GW) irá ignorar o campo</p>	<p>serial: 1</p> <p>tcp: 255 (0xFF)</p>	2
função	<p>Vazio ou valor de uma função suportada:</p> <p>1 - Ler bobina, 2 - Ler entrada discreta, 3 - Ler registros de retenção, 4 - Ler registros de entrada</p>	vazio	3
endereço	<p>Endereço do primeiro registro, bobina ou entrada.</p> <p>Se 'função' estiver vazio, então 'endereço' deverá estar no intervalo de: Coil - 00001 - 09999 Discreto entrada - 10001 - 19999 Registro de entrada - 30001 - 39999 Registro de retenção - 40001 - 49999</p> <p>Se 'função' não estiver vazio, o campo 'endereço' será de 0 a 65535 e usado sem modificação (PDU)</p>	<p>função vazia: 00001</p> <p>função não vazia: 0</p>	9999
count	<p>Contagem de 'tipo' sequenciado que será lido do dispositivo, onde:</p> <p>para Coil ou Discrete insira o 'type' = 1 bit para outros casos: $(count * type) / 2$ = contagem real de registros para leitura</p> <p>Se 'offset' não for 0, o valor será adicionado a 'contagem real' O intervalo aceitável para 'contagem real' é 1:65535</p>	1	2

Parâmetro	Descrição	Padrões	Exemplo
type	<p>Tipo de dados:</p> <p>para Read Coil e Read Discrete Input - bit</p> <p>para Read Holding Registers e Read Input Registers:</p> <p>int8 - 8bit uint8 - 8 bits (sem sinal)</p> <p>int16 - 16 bits</p> <p>uint16 - 16 bits (sem sinal)</p> <p>int32 - 32 bits</p> <p>uint32 - 32 bits (sem sinal)</p> <p>float - 32bit</p> <p>uint64 - 64bit (sem sinal)</p> <p>double - 64bit</p>	<p>bit</p> <p>uint16</p>	uint64
endianness	<p>Tipo de Endianness:</p> <p>be - Big Endian</p> <p>le - Little Endian</p> <p>mbe - Mid-Big Endian</p> <p>mle - Mid-Little Endian</p> <p>Limitações:</p> <p>para 1 bit - be</p> <p>para 8 bits - be,le</p> <p>para 16 bits - be,le</p>	be	le
offset	<p>Número de registradores, a partir de 'endereço', cujo resultado será descartado.</p> <p>O tamanho de cada registrador é de 16 bits (necessário para suportar equipamentos que não suportam acesso aleatório de leitura).</p>	0	4

16 Criando nomes personalizados para contadores de desempenho no VMware

Overview

The VMware performance counter path has the `group/counter[rollup]` format where:

- `group` - the performance counter group, for example `cpu`
- `counter` - the performance counter name, for example `usagemhz`
- `rollup` - the performance counter rollup type, for example `average`

So the above example would give the following counter path: `cpu/usagemhz[average]`

Overview

O caminho do contador de desempenho do VMware tem o `group/counter[rollup]` onde:

- `group` - o grupo do contador de desempenho, por exemplo `cpu`
- `counter` - o nome do contador de desempenho, por exemplo `usagemhz`
- `rollup` - o contador de desempenho tipo rollup, por exemplo `average`

Portanto, o exemplo acima resultaria no seguinte caminho do contador: `cpu/usagemhz[average]`

As descrições do contador de desempenho grupo, nomes e tipos rollup podem ser encontrados em [documentação VMware](#).

É possível obter nomes internos e criar nomes personalizados para contadores de desempenho usando o item de script no Zabbix.

It is possible to obtain internal names and create custom performance counter names by using script item in Zabbix.

Configuração

1. Crie um item de script desabilitado no host principal do VMware (onde o item **eventlog[]** está presente) com os seguintes parâmetros:

Item

Tags

Preprocessing

*

 Name

VMware metrics

Type

Script

*

 Key

vmware.metrics

Type of information

Text

Parameters

Name

Value

Add

*

 Script

try {...

*

 Timeout

10s

*

 Update interval

1m

Custom intervals

Type

Flexible

Scheduling

Interval

50s

Period

1-7,00:00-24

Add

*

 History storage period

Do not keep history

Storage period

Populates host inventory field

-None-

Description

Enabled

☐

Add

Test

Cancel

- Nome: VMware metrics
- Tipo: Script
- Key: vmware.metrics
- Tipo de informação: Text
- Script: copie e cole o **script** fornecido abaixo
- Tempo limite: 10
- Período de armazenamento de histórico: Não manter histórico
- Habilitado: não selecionado

Script

```
try {  
  Zabbix.log(4, 'vmware metrics script');  
  
  var result, resp,
```

```

req = new HttpRequest();
req.addHeader('Content-Type: application/xml');
req.addHeader('SOAPAction: "urn:vim25/6.0"');

login = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:vim25/6.0">
<soapenv:Header/>
<soapenv:Body>
  <urn:Login>
    <urn:_this type="SessionManager">SessionManager</urn:_this>
    <urn:userName>{$VMWARE.USERNAME}</urn:userName>
    <urn:password>{$VMWARE.PASSWORD}</urn:password>
  </urn:Login>
</soapenv:Body>
</soapenv:Envelope>'
resp = req.post("{$VMWARE.URL}", login);
if (req.getStatus() != 200) {
  throw 'Response code: '+req.getStatus();
}

query = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:vim25/6.0">
<soapenv:Header/>
<soapenv:Body>
  <urn:RetrieveProperties>
    <urn:_this type="PropertyCollector">propertyCollector</urn:_this>
    <urn:specSet>
      <urn:propSet>
        <urn:type>PerformanceManager</urn:type>
        <urn:pathSet>perfCounter</urn:pathSet>
      </urn:propSet>
      <urn:objectSet>
        <urn:obj type="PerformanceManager">PerfMgr</urn:obj>
      </urn:objectSet>
    </urn:specSet>
  </urn:RetrieveProperties>
</soapenv:Body>
</soapenv:Envelope>'
resp = req.post("{$VMWARE.URL}", query);
if (req.getStatus() != 200) {
  throw 'Response code: '+req.getStatus();
}
Zabbix.log(4, 'vmware metrics=' + resp);
result = resp;

logout = '<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:vim25/6.0">
<soapenv:Header/>
<soapenv:Body>
  <urn:Logout>
    <urn:_this type="SessionManager">SessionManager</urn:_this>
  </urn:Logout>
</soapenv:Body>
</soapenv:Envelope>'

resp = req.post("{$VMWARE.URL}", logout);
if (req.getStatus() != 200) {
  throw 'Response code: '+req.getStatus();
}

} catch (error) {
  Zabbix.log(4, 'vmware call failed : '+error);
  result = {};
}

return result;

```

Uma vez que o item está configurado, clique no botão Test, e então clique em Get value.

Test item



Get value from host ☒

Host address

Port

Proxy

Get value

Copie o XML recebido para qualquer formatador de XML e encontre a métrica desejada.

Um exemplo de XML para uma métrica:

```
<PerfCounterInfo xsi:type="PerfCounterInfo">
  <key>6</key>
  <nameInfo>
    <label>Usage in MHz</label>
    <summary>CPU usage in megahertz during the interval</summary>
    <key>usagemhz</key>
  </nameInfo>
  <groupInfo>
    <label>CPU</label>
    <summary>CPU</summary>
    <key>cpu</key>
  </groupInfo>
  <unitInfo>
    <label>MHz</label>
    <summary>Megahertz</summary>
    <key>megaHertz</key>
  </unitInfo>
  <rollupType>average</rollupType>
  <statsType>rate</statsType>
  <level>1</level>
  <perDeviceLevel>3</perDeviceLevel>
</PerfCounterInfo>
```

Use o XPath para extrair o caminho do contador do XML recebido. Para o exemplo acima, o XPath será:

field	xPath	value
group	//groupInfo[../key=6]/key	cpu
counter	//nameInfo[../key=6]/key	usagemhz
rollup	//rollupType[../key=6]	average

O caminho resultante do contador de desempenho neste caso é: `cpu/usagemhz[average]`

6 Supported functions

Click on the respective function group to see more details.

Function group	Functions
Aggregate func-tions	avg, bucket_percentile, count, histogram_quantile, item_count, kurtosis, mad, max, min, skewness, stddevpop, stddevsamp, sum, sumofsquares, varpop, varsamp
Foreach functions	avg_foreach, bucket_rate_foreach, count_foreach, exists_foreach, last_foreach, max_foreach, min_foreach, \sum_foreach
Bitwise func-tions	bitand, bitlshift, bitnot, bitor, bitrshift, bitxor

Function group	Functions
Date and time functions	date, dayofmonth, dayofweek, now, time
History functions	change, changecount, count, countunique, find, first, fuzzytime, last, logeventid, logseverity, logsource, monodec, monoinc, nodata, percentile, rate
Trend functions	baselinedev, baselinewma, trendavg, trendcount, trendmax, trendmin, trendstl, trendsum
Mathematical functions	abs, acos, asin, atan, atan2, avg, cbrt, ceil, cos, cosh, cot, degrees, e, exp, expm1, floor, log, log10, max, min, mod, pi, power, radians, rand, round, signum, sin, sinh, sqrt, sum, tan, truncate
Operator functions	between, in
Prediction functions	forecast, timeleft
String functions	ascii, bitlength, bytelength, char, concat, insert, left, length, ltrim, mid, repeat, replace, right, rtrim, trim

These functions are supported in **trigger expressions** and **calculated items**.

Foreach functions are supported only for **aggregate calculations**.

1 Funções agregadas

Exceto onde indicado de outra forma, todas as funções listadas aqui são suportadas dentro:

- **Expressões de gatilho**
- **Itens calculados**

As funções agregadas podem funcionar com:

- histórico de itens, por exemplo, `min(/host/key, 1h)`
- **funções foreach** como o único parâmetro, por exemplo, `min(last_foreach(/*/key))`

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >
- Os parâmetros específicos da função são descritos com cada função
- Os parâmetros `/host/key` e `(sec|#num)<:time shift>` nunca devem ser citado

Parâmetros comuns

- `/host/key` é um primeiro parâmetro obrigatório comum para as funções referenciando o histórico de itens do host
- `(sec|#num)<:time shift>` é um segundo parâmetro comum para o funções que fazem referência ao histórico do item do host, onde:
 - **sec** - máximo **avaliação period** em segundos (tempo **suffixos** podem ser usados), ou
 - **#num** - máximo **avaliação range** na última valores coletados (se precedidos por uma marca de hash)
 - **time shift** (opcional) permite mover o ponto de avaliação volta no tempo. Ver mais detalhes[</manual/config/triggers/expression#> em especificando a mudança de horário.

Funções agregadas

FUNÇÃO

	Descrição	Parâmetros específicos da função	Comentários
avg (/host/key,(sec #num)<:time shift>)			

Valor
médio de
um item
dentro do
período
de
avaliação
definido.

Consulte
**parâmet-
ros
comuns.**

Tipos de
valor su-
portados:
float, int

Exemplos:
=> **

avg(/host/key,1h)

→ **valor
médio da
última
hora até
agora**
=>

**avg(/host/
key,1h:now-
1d)** →

**valor
médio de
uma
hora de
25 horas
atrás a
24 horas
atrás de
agora**
=>

média**
(/host/key,#5)

→ valor
médio dos
cinco
últimos
valores
=>

**avg(/host/key,#5:now-
1d)** →

valor
médio dos
cinco
últimos
valores
excluindo
os valores
recebidos
nas
últimas
24 horas

A
mudança
de horário
é útil
quando
há neces-
sidade de
comparar
o valor
médio
atual com
o valor
médio há
algum
tempo.

FUNÇÃO

bucket_percentile (filtro de item, período de tempo, porcentagem)

Calcula o percentil dos intervalos de um histograma.	filtro de item - consulte filtro de item período de tempo - consulte período de tempo percentage - porcentagem (0-100)	Suportado apenas em itens calculados. Esta função é um alias para histogram_quantile bucket_rate_foreach filter, time period, 1))
--	---	---

count (func_foreach(filtro de item,<período de tempo>))

Contagem de valores em uma matriz retornada por uma função foreach.	func_foreach - função foreach para a qual o número de valores retorna-dos deve ser contado (com argumentos suportados). Consulte funções foreach para obter detalhes.	func_foreach Tipo de valor suportado: int Exemplo: => count (max_foreach(/net.if.in[*],1h)) → número de itens net.if.in que receberam dados na última hora até agora > Observe que usar count() com uma função foreach relacionada ao histórico (max_foreach, avg_foreach etc.) pode levar a implicações de desempenho, enquanto usar exists_foreach() , que funciona apenas com dados de configuração, não terá esse efeito.
histogram_quantile (quantil,bucket1,value1,bucket2,value2,...)		

	Calcula o ϕ -quantil dos buckets de um histograma.	quantil - $0 \leq \phi \leq 1$ bucketN , valueN - pares inseridos manualmente (≥ 2) de parâmetros ou resposta de bucket_rate_filter	Suportado apenas em itens calculados. Funcionalmente corresponde a <code>'histogram_quantile'</code> do PromQL. Retorna -1 se os valores do último bucket <code>'Infinity'</code> (<code>" + inf"</code>) forem iguais a 0. br> Exemplo: => histograma_quantil (0.75, _bucket[Inf]) => histograma_quantile (0,5,
item_count (filtro de item)	Contagem de itens existentes na configuração que correspondem aos critérios de filtro.	filtro de item - critério para seleção de item, permite referência por grupo de host, host, chave de item e tags. Caracteres curinga são suportados. Consulte filtro de item para obter mais detalhes.	Tipo de valor suportado: int Funciona como um alias para <code>*count(exists_foreach(item))*</code> função. Exemplo: => item_count (/*/agent.pir group 1"]]) → número de hosts com o item <code>agent.ping</code> no <code>"Host group 1"</code> kurtosis (/host/key,(sec #num)<:time shift>)

	<p>"Tailedness" da distribuição de probabilidade em valores coletados dentro do período de avaliação definido.</p> <p>Veja também: Curtosis</p>	<p>Consulte common parameters.</p>	<p>Tipos de valor suportados: float, int</p> <p>Exemplo: => curtose(/host/key,1h) → curtose para a última hora até agora</p>
Mad (/host/key,(sec #num)<:time shift>)	<p>Desvio absoluto médio em valores coletados dentro do período de avaliação definido.</p> <p>Veja também: Desvio absoluto médio</p>	<p>Consulte parâmetros comuns.</p>	<p>Tipos de valor suportados: float, int</p> <p>Exemplo: => mad(/host/key,1h) → desvio absoluto médio para a última hora até agora</p>
max (/host/key,(sec #num)<:time shift>)	<p>Valor mais alto de um item dentro do período de avaliação definido.</p>	<p>Consulte parâmetros comuns.</p>	<p>Tipos de valor suportados: float, int</p> <p>Exemplo: => ** max(/host/key,1h) - min(/host/key,1h**) → calcular a diferença entre os valores máximo e mínimo na última hora até agora</p>
min (/host/key,(sec #num)<:time shift>)			

	Menor valor de um item dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Exemplo: => ** max(/host/key,1h) - min(/host/key,1h**) → calcular a diferença entre os valores máximo e mínimo na última hora até agora (delta de valores)
assimetria (/host/key,(sec #num)<:time shift>)	Assimetria da distribuição de probabilidade em valores coletados dentro do período de avaliação definido.	Veja parâmetros comuns (#common-parameters). Exemplo: => assimetria(/host/key,1h) → assimetria para o último hora até agora	Tipos de valor suportados: float, int
	Veja também: Skewness		
stddevpop (/host/key,(sec #num)<:time shift>)	Desvio padrão da população em valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Exemplo: => stddevpop(/host/key,1h) → desvio padrão da população para o último hora até agora
	Veja também: Desvio padrão		
stddevsamp (/host/key,(sec #num)<:time shift>)			

	Desvio padrão da amostra em valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Pelo menos dois valores de dados são necessários para que esta função funcione. Exemplo: => <code>**std-devsamp* (/host/key,1h)</code> → desvio padrão da amostra para a última hora até agora
soma (/host/key,(sec #num)<:time shift>)	Soma dos valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Exemplo: => soma (/host/key,1h) → soma dos valores da última hora até agora
sumofsquares (/host/key,(sec #num)<:time shift>)	A soma dos quadrados em valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Exemplo: => sumofsquares (/host/key,1h) → soma dos quadrados da última hora até agora
varpop (/host/key,(sec #num)<:time shift>)			

FUNÇÃO

	Variação populacional dos valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Exemplo: => varpop (/host/key, 1h) → variação de população para a última hora até agora
varsamp (/host/key,(sec #num)<:time shift>)	Variação de amostra de valores coletados dentro do período de avaliação definido.	Consulte parâmetros comuns .	Tipos de valor suportados: float, int Pelo menos dois valores de dados são necessários para que esta função funcione. Exemplo: => varsamp (/host/key, 1h) → variação de amostra para a última hora até agora

1 Funções Foreach

Visão geral

As funções foreach retornam valores agregados do histórico de vários Itens. As funções foreach são usadas em cálculos agregados.

As seguintes funções foreach são suportadas:

Função	Descrição
avg_foreach	Retorna valores médios
bucket_rate_foreach	Retorna pares (limite superior do bucket, valor da taxa) adequados para uso na função histogram_quantile() , onde "limite superior do bucket " é o valor do parâmetro-chave do item definido pelo <número do parâmetro> parâmetro .

Função	Descrição
count_foreach	Retorna o número de valores
exists_foreach	Retorna o número de itens atualmente habilitados
last_foreach	Retorna os últimos valores
max_foreach	Retorna os valores máximos
min_foreach	Retorna valores mínimos
soma_foreach	Retorna a soma dos valores

As funções foreach retornam uma matriz de valores - um para cada item. Itens são selecionados usando filtros no primeiro parâmetro.

Veja [agregar cálculos](#) para mais detalhes sobre o uso de funções foreach.

Supported functions

Function	Description
avg_foreach	Returns the average value for each item.
bucket_rate_foreach	Returns pairs (bucket upper bound, rate value) suitable for use in the histogram_quantile() function, where "bucket upper bound" is the value of item key parameter defined by the <parameter number> parameter .
count_foreach	Returns the number of values for each item..
exists_foreach	Returns the number of currently enabled items.
last_foreach	Returns the last value for each item.
max_foreach	Returns the maximum value for each item.
min_foreach	Returns the minimum value for each item.
sum_foreach	Returns the sum of values for each item.

Parâmetros

As funções Foreach suportam dois parâmetros comuns - filtro de item e tempo período:

foreach_function (filtro de item, período de tempo)

por exemplo:

avg_foreach(/*/mysql.qps?[group="MySQL Servers"],5m)

Algumas funções suportam parâmetros adicionais.

Filtro de itens

O parâmetro **first** permite filtrar os itens que nos interessam. Um filtro complexo pode ser usado, referenciando a chave do item, grupo de hosts e tags, conforme ilustrado pelos exemplos:

Exemplo de sintaxe	Descrição
/host/key[abc,*]	Corresponde a itens semelhantes neste host.
*/key	Corresponde ao mesmo item de qualquer host.
*/key?[group="ABC" and tag="tagname:value"]	Corresponde ao mesmo item de qualquer host do grupo ABC com tags 'tagname:value'.
/key[a,,c]?[(grupo="ABC" e tag="Tag1") ou (grupo="DEF" e (tag="Tag2" ou tag=" Tag3:value"))]	Corresponde a itens semelhantes de qualquer host do grupo ABC ou DEF com as respectivas tags.

Todos os itens referenciados devem existir e coletar dados. Apenas itens ativados em hosts habilitados são incluídos nos cálculos.

::: não importante Se a chave de item de um item referenciado for alterada, o filtro deve ser atualizado manualmente. :::

Especificar um grupo de hosts pai inclui o grupo pai e todos os grupos de hosts com seus itens.

Período de tempo

O parâmetro **second** permite especificar o período de tempo para agregação. O período de tempo só pode ser expresso como tempo, a quantidade de valores (prefixado com #) não é suportado.

Símbolos de unidade suportados podem ser usados neste parâmetro por conveniência, por exemplo '5m' (cinco minutos) em vez de '300s' (300 segundos) ou '1d' (um dia) em vez de '86400' (86400 segundos).

O período de tempo é ignorado pelo servidor se passado com last_foreach função e pode, portanto, ser omitido:

```
last_foreach(/*/key?[group="host group"])
```

O período de tempo não é compatível com a função `exists_foreach`.

Parâmetros adicionais

Um terceiro parâmetro opcional é suportado pelo `bucket_rate_foreach` função:

```
bucket_rate_foreach(filtro de item, período de tempo,<número do parâmetro>)
```

onde <número do parâmetro> é a posição do valor "bucket" em a chave do item. Por exemplo, se o valor "bucket" em `myItem[aaa,0.2]` for '0.2', então sua posição é 2.

O valor padrão de <número do parâmetro> é '1'.

Behavior depending on availability

The following table illustrates how each function behaves in cases of limited availability of host/item and history data.

Function	Disabled host	Unavailable host with data	Unavailable host without data	Disabled item	Unsupported item	Data retrieval error (SQL)
avg_foreach	ignore	return avg	ignore	ignore	ignore	ignore
bucket_rate_foreach	ignore	return bucket rate	ignore	ignore	ignore	ignore
count_foreach	ignore	return count	0	ignore	ignore	ignore
exists_foreach	ignore	1	1	ignore	1	n/a
last_foreach	ignore	return last	ignore	ignore	ignore	ignore
max_foreach	ignore	return max	ignore	ignore	ignore	ignore
min_foreach	ignore	return min	ignore	ignore	ignore	ignore
sum_foreach	ignore	return sum	ignore	ignore	ignore	ignore

If the item is ignored, nothing is added to the aggregation.

2 Funções bit a bit

Todas as funções listadas aqui são suportadas em:

- **Expressões de gatilho**
- **Itens calculados**

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Expressões são aceitas como parâmetros
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >

FUNÇÃO

Descrição	Parâmetros específicos da função	Comentários
-----------	----------------------------------	-------------

bitand (valor,máscara)

FUNÇÃO

Valor de "E bit a bit" de um valor de item e máscara.	valor - valor a ser verificado máscara (obrigatório) - inteiro sem sinal de 64 bits (0 - 18446744073709551615)	Suportado tipos de valor: int Embora a comparação seja feita de forma bit a bit, os valores devem ser fornecidos e retornados em decimal. Por exemplo, a verificação do 3º bit é feita comparando-se a 4, não a 100. Exemplos: => bitand (last(/host/key), 12)=8 ou bitand (last(/host/key), 12) → 3º ou 4º bit definido, mas não ambos ao mesmo tempo => bitand (last(/host/key), 20)=16 → 3º bit não definido e 5º bit definido.
bitlshift (valor, bits para deslocamento)		

	Deslocamento bit a bit para a esquerda de um valor de item.	valor - valor a ser verificado bits para desloca- mento (obri- gatório) - número de bits para desloca- mento	Tipos de valor su- portados: int < br>Embora a com- paração seja feita de forma bit a bit, todos os valores devem ser forneci- dos e são retorna- dos em decimal. Por exemplo, a verifi- cação do 3º bit é feita comparando- se a 4, não a 100.
bitnot (valor)	Valor de "NÃO bit a bit" de um valor de item.	valor - valor a ser verificado	Tipos de valor su- portados: int Embora a compara- ção seja feita de maneira bit a bit, todos os valores devem ser forneci- dos e são retorna- dos em decimal. Por exemplo, a verifi- cação do 3º bit é feita comparando- se a 4, não a 100.

FUNÇÃO

bitor (valor,máscara)

Valor de "OR bit a bit" de um valor de item e máscara.	valor - valor a ser verificado máscara (obrigatório) - inteiro sem sinal de 64 bits (0 - 18446744073709551615)	Suportado tipos de valor: int Embora a comparação seja feita de forma bit a bit, todos os valores devem ser fornecidos e retornados em decimal. Por exemplo, a verificação do 3º bit é feita comparando-se a 4, não a 100.
--	---	---

bitrshift (valor, bits para deslocamento)

Deslocamento bit a bit para a direita de um valor de item.	valor - valor a ser verificado bits a serem deslocados (obrigatório) - número de bits a serem deslocados	Tipos de valor suportados: int Embora a comparação seja feita de forma bit a bit, todos os valores devem ser fornecidos e são retornados em decimal. Por exemplo, a verificação do 3º bit é feita comparando-se a 4, não a 100.
--	---	--

FUNÇÃO

bitxor (valor,máscara)

Valor de "OR exclusivo bit a bit" de um valor de item e máscara.	valor - valor a ser verificado máscara (obrigatório) - inteiro sem sinal de 64 bits (0 - 18446744073709551615)	Tipos de valores suportados: int Embora a comparação seja feita de forma bit a bit, todos os valores devem ser fornecidos e são retornados em decimal. Por exemplo, a verificação do 3º bit é feita comparando-se a 4, não a 100.
--	---	--

3 Funções de data e hora

Todas as funções listadas aqui são suportadas em:

- Expressões de gatilho
- Itens calculados

::: não importante As funções de data e hora não podem ser usadas na expressão sozinha; pelo menos uma função não baseada em tempo referenciando o item de host deve estar presente na expressão. :::

FUNÇÃO

data

Descrição	Parâmetros específicos da função	Comentários
-----------	----------------------------------	-------------

Data atual no formato AAAAM-MDD.		Exemplo: => data() <20220101
----------------------------------	--	-------------------------------------

diadomês

Dia do mês no intervalo de 1 a 31.		Exemplo: => dayof-month() =1
------------------------------------	--	-------------------------------------

dia da semana

FUNÇÃO		
agora	Dia da semana no intervalo de 1 a 7 (Seg - 1, Dom - 7).	Exemplo: => day-ofweek() <6
	Número de segundos desde a época (00:00:00 UTC, 1º de janeiro de 1970).	Exemplo: => agora() <1640998800
hora	Hora atual no formato HHMMSS.	Exemplo: => hora() >000000 e hora() <060000

4 Funções de histórico

Todas as funções listadas aqui são suportadas em:

- **Expressões de gatilho**
- **Itens calculados**

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >
- Os parâmetros específicos da função são descritos com cada função
- Os parâmetros `/host/key` e `(sec|#num)<:time shift>` nunca devem ser citado

Parâmetros comuns

- `/host/key` é um primeiro parâmetro obrigatório comum para as funções referenciando o histórico de itens do host
- `(sec|#num)<:time shift>` é um segundo parâmetro comum para o funções que fazem referência ao histórico do item do host, onde:
 - **sec** - máximo **avaliação period** em segundos (tempo **sufixos** podem ser usados), ou
 - **#num** - máximo **avaliação range** na última valores coletados (se precedidos por uma marca de hash)
 - **time shift** (opcional) permite mover o ponto de avaliação volta no tempo. Ver mais detalhes[/manual/config/triggers/expression#] em especificando a mudança de horário.

Funções do histórico

FUNÇÃO		
	Descrição	Parâmetro específico - função
baselinedev	(/host/key,data period:time shift,season_unit,num_seasons)	

Retorna o número de desvios (pelo algoritmo stddevpop) entre o último período de dados e os mesmos períodos de dados nas temporadas anteriores	data period -o período de coleta de dados dentro de uma estação, definido como <N><time unit> onde N - número de unidades de tempo time unit - h (hora), d (dia), w (semana), M (mês) ou y (ano), deve ser igual ou menor que a estação Time shift - o deslocamento do período de tempo (veja exemplos) season_unit - duração de uma temporada (h, d, w, M, y), não pode ser menor que o período de dados num_seasons - número de temporadas a avaliar	Exemplos: => baselinewma (/host/key,1d:nomeo padrão → calcular os números de desvios padrão (população) entre o dia anterior e o mesmo dia nos 6 meses anteriores. Caso a data não exista em um mês anterior, será utilizado o último dia do mês (31/jul será analisado contra 31/jan, 28/fev,... 30/junho). => baselinewma (/host/key,1h:nomeo padrão → calculando o número de desvios padrão (população) entre a hora anterior e as mesmas horas no período de dez dias antes de ontem.
--	---	---

baselinewma (/host/key,data period:time shift,season_unit,num_seasons)

Calcula a linha de base calculando a média dos dados do mesmo período de tempo em vários períodos iguais ('estações') usando o algoritmo de média móvel ponderada.	data period - o período de coleta de dados dentro de uma temporada, definido como N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano), deve ser igual ou menor que a temporada	Exemplos: => base-linewma (/host/key,1h:n → cálculo da linha de base com base na última hora completa dentro de um período de 3 dias que terminou ontem. Se "agora" for 13:30 de segunda-feira, os dados de 12:00-12:59 de sexta, sábado e domingo serão analisados. => base-linemwa (/host/key,2h:n → calcular a linha de base com base nas últimas duas horas em um período de 3 dias que terminou ontem. Se "agora"
Time shift -o deslocamento do período de tempo, define o fim do período de coleta de dados em temporadas (veja exemplos)		
season_unit for - duração de uma temporada (h, d, w, M, y), não pode ser menor que o período de dados		segunda-feira 13h30, os dados para 10h-11h59 de sexta, sábado e domingo serão analisados.
num_seasons -número de temporadas a		=> base-linemwa (/host/key,1d:n → cálculo

FUNÇÃO

change (/host/key)

A quantidade de diferença entre o valor anterior e o mais recente.

Tipos de valor suportados: float, int, str, text, log

Para strings retorna:
0 - os valores são iguais
1 - os valores diferem

Exemplo:
=> **alterar**(/host/key)>10

Numeric difference will be calculated, as seen with these incoming example values ('previous' and 'latest' value = difference):
'1' and '5' = +4
'3' and '1' = -2
'0' and '-2.5' = -2.5

See also:
abs for comparison

changecount (/host/key,(sec|#num)<:time shift>,<mode>)

Número de alterações entre valores adjacentes dentro do período de avaliação definido.	<p>Consulte Parâmetros comuns.</p> <p>mode (opcional; deve estar entre aspas duplas)</p> <p>Modos compatíveis: all - conta todas as alterações (padrão) dec - contagem diminui inc - contagem aumenta</p>	<p>Tipos de valor suportados: float, int, str, text, log</p> <p>Para tipos de valor não numéricos, o parâmetro mode é ignorado.</p> <p>Exemplos: => ** change-count(/host/key, 1w) → número de mudanças de valor para a última semana até agora
=> change-count(/host/key,#10,"i") → número de aumentos de valor (em relação ao valor adjacente) entre os últimos 10 valores
=> change-count**(/host/key,24h,"c") → número de valores diminui (em relação ao valor adjacente) nas últimas 24 horas até agora</p>
--	---	---

count (/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Número de valores dentro do período de avaliação definido.	Veja parâmetros comuns.	Tipos de valor suportados:
	operator (opcional; deve estar entre aspas duplas)	float, integer, string, text, log Os itens flutuantes correspondem com a precisão de 2.22e-16; se o banco de dados for não atualizado a precisão é de 0.000001.
	Operadores suportados: eq - igual (padrão) ne - diferente gt - maior ge - maior ou igual lt - menor le - menor ou igual como - corresponde se contiver padrão (diferencia maiúsculas de minúsculas) bitand - bit a bit AND regexp - correspondência com distinção entre maiúsculas e minúsculas da expressão regular fornecida em pattern iregexp - correspondência sem distinção entre maiúsculas e minúsculas de a	Com bitand como terceiro parâmetro, o quarto parâmetro pattern pode ser especificado como dois números, separados por '/': number_to_compare_with count() calcula "and bit a bit" a partir do valor e da mask e compara o resultado com number_to_compare_with. Se o resultado de "bit a bit AND" for igual a number_to_compare_with, o valor será contado. Se number_to_compare_with

FUNÇÃO

countunique (/host/key,(sec|#num)<:time shift>,<operator>,<pattern>)

Número de valores únicos dentro do período de avaliação definido.	Consulte parâmetros comuns . operator (opcional; deve estar entre aspas duplas) Operadores suportados: eq - igual (padrão) ne - diferente gt - maior ge - maior ou igual lt - menor le - menor ou igual like - corresponde se contiver padrão (diferencia maiúsculas de minúsculas) bitand - bit a bit AND regex - correspondência com distinção entre maiúsculas e minúsculas da expressão regular fornecida em <code>pattern</code> iregexp - correspondência sem distinção entre maiúsculas e minúsculas da expressão	Tipos de valor suportados: float, integer, string, text, log Float os itens correspondem com a precisão de 2,22e-16; se o banco de dados for não atualizado a precisão é 0,000001. Com <code>bitand</code> como terceiro parâmetro, o quarto parâmetro <code>pattern</code> pode ser especificado como dois números, separados por <code>'/'</code> : number_to_compare_with . <code>count()</code> calcula "and bit a bit" a partir do valor e da <code>mask</code> e compara o resultado com <code>number_to_compare_with</code> . Se o resultado de "bit a bit AND" for igual a <code>number_to_compare_with</code> , o valor será contado. Se <code>number_to_compare_with</code>
---	---	---

FUNÇÃO

encontrar (/host/key,<(sec|#num)<:time shift>,<operator>,<pattern>)

Encontre uma correspondência de valor.	Consulte parâmetros comuns . sec ou #num (opcional) - o padrão é o valor mais recente se não especificado operador (opcional; deve estar entre aspas duplas) "operadores"um valor suportados: eq - igual (padrão) ne - diferente gt - maior ge - maior ou igual lt - menor le - menor ou igual like - valor contém a string dado em <code>pattern</code> (diferença maiúsculas de minúsculas) bitand - bit a bit AND regexp - correspondência com distinção entre maiúsculas e minúsculas da expressão regular fornecida em <code>pattern</code>	Tipos de valor suportados: float, int, str, text, log Retorna: 1 - encontrado 0 - caso contrário Se mais de um valor for processado, '1' será retornado se houver pelo menos um valor correspondente. Com regexp ou iregexp como o terceiro parâmetro, o o quarto parâmetro <code>pattern</code> pode ser uma expressão regular comum ou global (começando com '@'). No caso de expressões regulares globais, a distinção entre maiúsculas e minúsculas é herdada das configurações de expressões regulares globais. Exemplo:
--	--	---

FUNÇÃO

primeiro (/host/key,sec<:time shift>)

O
primeiro
(o mais
antigo)
valor
dentro do
período
de
avaliação
definido.

Consulte
**parâmet-
ros
comuns.**

Tipos de
valor su-
portados:
float, int,
str, text,
log

Exemplo :
=>
first(/host/key,**1h**)
→
recuperar
o valor
mais
antigo na
última
hora até
agora
< br>Veja
também
last().

fuzzytime (/host/chave,seg)

Verificando o quanto o tempo do agente passivo difere do tempo do servidor/proxy Zabbix.	Consulte common-parameters .	<p>Tipos de valor suportados: float, int</p> <p>Retorna:</p> <p>1 - diferença entre o valor do item passivo (como carimbo de data/hora) e o carimbo de data/hora do servidor/proxy Zabbix é menor ou igual a T segundos</p> <p>0 - caso contrário</p> <p>Geralmente usado com o item 'system.localtime' para verificar esse local a hora está sincronizada com a hora local do servidor Zabbix. Observe que 'system.localtime' deve ser configurado como uma verificação passiva. Pode ser usado também com vfs.file.time[/path/file,modify] para verificar se o arquivo</p>
--	-------------------------------------	---

FUNÇÃO

último (/host/key,<#num<:time shift»)

O valor
mais
recente.

Consulte
**parâmet-
ros
comuns.**

#num
(opcional)

- o
enésimo
valor mais
recente

Tipos de
valor su-
portados:
float, int,
str , text,
log

Observe
que um
período
de tempo
com tag
de hash
(#N)
funciona
de
maneira
diferente
aqui do
que com
muitas
outras
funções.
Por
exemplo:
last() é
sempre
igual a
last (#1)
last(#3) -
terceiro
valor mais
recente
(não três
valores
mais
recentes)

O Zabbix
não
garante a
ordem
exata dos
valores se
existirem
mais de
dois
valores
em um
segundo
na
história.

Exemplo:

=>

last(/host/key)

→

recuperar
o último
valor

=>

**last(/host/key,*
*#2) →**

**recu-
perar o**

FUNÇÃO

logeventid (/host/key,<#num<:time shift>,<pattern>)

Verificando se o ID do evento da última entrada de log corresponde a uma expressão regular.	Consulte parâmetros comuns . #num (opcional) - o enésimo valor mais recente pattern (opcional) - expressão regular que descreve o padrão requerido, estilo Perl Compatible Regular Expression (PCRE) (os argumentos de string devem ser aspas duplas).	Tipos de valor suportados: log Retorna: 0 - não corresponde 1 - corresponde
---	---	--

logseverity (/host/key,<#num<:time shift>)

FUNÇÃO

Gravidade do log da última entrada de log.	Consulte parâmetros comuns.	Tipos de valor suportados : log
	#num (opcional) - o enésimo valor mais recente	Retorna: 0 - gravidade padrão N - gravidade (inteiro, útil para logs de eventos do Windows: 1 - Informações, 2 - Aviso, 4 - Erro, 7 - Auditoria de falhas, 8 - Auditoria de sucesso, 9 - Crítico, 10 - Detalhado). O Zabbix obtém a gravidade do log do campo Informações do log de eventos do Windows.

logsource (/host/key,<#num<:time shift>,<pattern>)

Verificando se a origem do log da última entrada de log corresponde a uma expressão regular.	Consulte parâmetros comuns . #num (opcional) - o enésimo número mais valor recente padrão (opcional) - expressão regular que descreve o padrão necessário, estilo Perl Compatible Regular Expression (PCRE) (argumentos de string devem estar entre aspas duplas).	Tipos de valor suportados: log Retorna: 0 - não corresponde 1 - corresponde Normalmente usado para logs de eventos do Windows. Por exemplo, log-source("VMware Server").
--	--	--

monodec (/host/key,(sec|#num)<:time shift>,<mode>)

Verifique se houve uma diminuição monótona nos valores.

Consulte **parâmetros comuns.**

modo (deve estar entre aspas duplas) - fraco (todo valor é menor ou igual ao anterior; padrão) ou strict (todos os valores diminuíram)

Tipos de valor suportados: int

Retorna 1 se todos os elementos no período de tempo diminuem continuamente, 0 caso contrário.

Exemplo: => **mon-**

odec(/Host1/system.swap.size[all,free],**60s**) + **mon-odec**(/Host2/system.swap.size[all,free],**60s**) + **mon-odec**(/Host3/system.swap.size[all,free],**60s**)

- calcule em quantos hosts houve uma diminuição no tamanho do swap gratuito

monoinc (/host/key,(sec|#num)<:time shift>,<mode>)

FUNÇÃO

Verifique se houve um aumento monótono nos valores.

Consulte **parâmetros comuns**.

modo (deve ser entre aspas duplas) - fraco (cada valor é maior ou igual ao anterior; padrão) ou strict (todo valor aumentou)

Tipos de valor suportados: int

Retorna 1 se todos os elementos no período de tempo aumentarem continuamente, 0 caso contrário .

Exemplo: =>

monoinc(/Host1/system - verifique se o horário local do sistema está aumentando consistentemente

nodata (/host/chave,seg,<modo>)

Verificando se não há dados recebidos.	Consulte parâmetros comuns .	Todos os tipos de valor são suportados.
	O período de s não deve ser inferior a 30 segundos porque o processo de sincronização de histórico calcula apenas esta função a cada 30 segundos.	Retorna: 1 - se nenhum dado for recebido durante o período de tempo definido 0 - caso contrário
	nodata(/host/ host , proxy) não é permitido.	Desde o Zabbix 5.0, os gatilhos 'nodata' monitora- dos , Os proxy são, por padrão, sensíveis à disponibilidade do proxy - se o proxy ficar indisponível, os gatilhos 'nodata' não serão acionados imediatamente após uma conexão restaurada, mas pularão os dados para o período atrasado. Observe que, para proxies passivos, a supressão é ativada se a conexão for restaurada mais de 15 segundos e não
	mode - se definido como strict (entre aspas duplas), esta função será insensível à disponibilidade do proxy (consulte os comentários para obter detalhes).	

FUNÇÃO

percentil (/host/key,(sec|#num)<:time shift>,percentage)

P-ésimo percentil de um período, em que P (porcentagem) é especificado pelo terceiro parâmetro.

Consulte **parâmetros comuns**.

Tipos de valor suportados: float, int

porcentagem

- um valor flutuante
- ponto número entre 0 e 100 (inclusive) com até 4 dígitos após o ponto decimal

taxa (/host/key,sec<:time shift>)

Taxa média por segundo do aumento em um contador monotonicamente crescente dentro do período de tempo definido.

Consulte **parâmetros comuns**.

Tipos de valor suportados: float, int

Funcionalmente corresponde a **'rate'** do PromQL.

Exemplo:
=> **rate** (/host/key,**30s**)
→ Se o aumento monotônico em 30 segundos for 20, esta função retornará 0,67.

trendavg (/host/key, time period:time shift)

Média de valores de tendência dentro do período de tempo definido.	período de tempo - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trendcount (/host/key, 1h:now) → média para a hora anterior (por exemplo, 12:00-13:00) => trendcount (/host/key, 1h:now-1h) → média para duas horas atrás (11:00-12:00) => trendcount (/host/key, 1h:now-2h) → média de três horas atrás (10:00-11:00) Deslocamento de tempo - o deslocamento do período de tempo (veja exemplos) => trendcount (/host/key, 1M:now-1y) → média do mês anterior um ano atrás
trendcount (/host/key, time period:time shift)		

Número de valores de tendência recuperados com sucesso dentro do período de tempo definido.	período de tempo - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trend-count(/host/key,1h:now) → contar para a hora anterior (por exemplo, 12:00-13:00) => trend-count(/host/key,1h:now 1h) → contar para duas horas atrás (11:00-12:00) => trend-count(/host/key,1h:now 2h) → contar por três horas atrás (10:00-11:00) => trend-count(/host/key,1M:now 1y) → contagem para o mês anterior um ano atrás
	Time shift - o deslocamento do período de tempo (veja exemplos)	

trendmax (/host/key, time period:time shift)

O máximo em valores de tendência dentro do período de tempo definido.	período de tempo - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trend-max (/host/key, 1h:now/ → máximo para a hora anterior (por exemplo, 12:00-13:00) => trend-max (/host/key, 1h:now/ - trend-min (/host/key, 1h:now/h) → calcule a diferença entre os valores máximo e mínimo (delta de tendência) para a hora anterior (12:00-13:00) => **trend-max* *(/host/key, 1h:now/h-1h) → máximo para duas horas atrás (11:00-12:00) => trend-max (/host/key, 1h:agora/h-2h) → máximo para três horas atrás (10:00-11:00) => trend-max (/host/key, 1M:now-1y) → máximo para o mês anterior há um ano
---	--	--

FUNÇÃO

trendmin (/host/key, time period:time shift)

O mínimo em valores de tendência dentro do período de tempo definido.	período de tempo - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trend-min (/host/key, 1h:now/h) → mínimo para a hora anterior (por exemplo, 12:00-13:00) => trend-min (/host/key, 1h:now/h) - trend-min (/host/key, 1h:agora/h) → calcule a diferença entre os valores máximo e mínimo (delta de tendência) para a hora anterior (12:00-13:00) => trend-min (/host/key, 1h:now/h-1h) → mínimo para duas horas atrás (11:00-12:00) => trend-min (/host/key, 1h:now/h-2h) → mínimo para três horas atrás (10:00-11:00) => trend-min (/host/key,**1M:now/1y* *) → mínimo do mês anterior há um ano
---	--	--

FUNÇÃO

trendstl (/host/key,eval period:time shift,detection
period,season,<deviations>,<devalg>,<s_window>)

Retorna uma taxa de anomalia - um valor decimal entre 0 e 1 que é ((o número de valores de anomalia no período de detecção) / (número total de valores no período de detecção).	período de avaliação - o período de tempo que deve ser de-composto (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de unidades de tempo unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trend-stl (/host/key, 100h:now/h ,10h,2h) → analisando as últimas 100 horas de dados de tendência, encontre a taxa de anomalias para as 10 horas anteriores desse período, esperando que a periodicidade seja 2h, a os valores da série restante do período de avaliação são considerados anomalias se atingirem o valor de 3 desvios do MAD dessa série restante => trend-stl (/host/key, 100h:now ,10h,2h,2.1,"louco") → analisando o período das 100 horas anteriores de dados de tendência, começando a contar 10 horas atrás encontre a taxa de
	Time shift - o deslocamento do período de tempo (veja exemplos)	
	período de detecção - o período de tempo a partir do final do período de avaliação para o qual as anomalias são calculadas (mínimo '1h', não pode ser maior que o período de avaliação), definido como <N><	

FUNÇÃO

trendsum (/host/key,time period:time shift)

Soma dos valores de tendência dentro do período de tempo definido.	período de tempo - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - número de tempo unidades unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).	Exemplos: => trendsum (/host/key, 1h:now/ → soma para o hora anterior (por exemplo, 12:00-13:00) => trendsum (/host/key, 1h:now/1h) → soma de duas horas atrás (11:00-12:00) => trendsum (/host/key, 1h:now/2h) → soma de três horas atrás (10:00-11:00) => trendsum (/host/key, 1M:now/1y) → soma para o mês anterior um ano atrás
--	---	---

Time shift

- o deslocamento do período de tempo (veja exemplos)

5 Funções de tendência

As funções de tendência, ao contrário das **funções de histórico**, usam dados de **tendência** para cálculos.

As tendências armazenam valores agregados por hora. As funções de tendência usam essas médias horárias e, portanto, são úteis para análises de longo prazo.

Os resultados da função de tendência são armazenados em cache para que várias chamadas para a mesma função com os mesmos parâmetros busquem informações do banco de dados apenas uma vez. O cache da função de tendência é controlado pelo parâmetro do servidor **TrendCacheSize**.

As triggers que fazem referência a funções de tendência **somente** são avaliados uma vez no menor período de tempo na expressão. Por exemplo, uma trigger como

```
trendavg(/host/key,1d:now/d) > 1 or trendavg(/host/key2,1w:now/w) > 2
```

será avaliado uma vez por dia. Se a trigger contiver funções de tendência e histórico (ou baseadas em tempo), é calculado de acordo com os **princípios usuais**.

Todas as funções listadas aqui são suportadas em:

- Expressões de trigger
- Itens calculados

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Parâmetros de funções opcionais (ou partes de parâmetros) são indicados por < >
- Os parâmetros específicos da função são descritos com cada função
- Os parâmetros /host/key e time period:time shift nunca devem ser citados

Parâmetros comuns

- /host/key é um primeiro parâmetro obrigatório comum
- time period:time shift é um segundo parâmetro comum, onde:
 - **período de tempo** - o período de tempo (mínimo '1h'), definido como <N><unidade de tempo> onde N - o número de unidades de tempo, unidade de tempo - h (hora), d (dia), w (semana), M (mês) ou y (ano).
 - **time shift** - o deslocamento do período de tempo (veja exemplos de função)

Trend functions

FUNCTION	Description Function-specific parameters	Comments
baselinedev (/host/key,data period:time shift,season_unit,num_seasons)		

Returns the number of deviations (by stddevpop algorithm) between the last data period and the same data periods in preceding seasons.	data period - the data gathering period within a season, defined as <N><time unit> where N - number of time units time unit - h (hour), d (day), w (week), M (month) or y (year), must be equal to or less than season	Examples: => base-linedev (/host/key,1d:no → calculating the number of standard deviations (population) between the previous day and the same day in the previous 6 months. If the date doesn't exist in a previous month, the last day of the month will be used (Jul,31 will be analysed against Jan,31, Feb, 28,... June, 30). => base-season_unitlinedev (/host/key,1h:no → calculating the number of standard deviations (population) between the previous hour and the same hours over the period of ten days before yesterday.
Time shift - the time period offset (see examples)		
num_seasons - number of seasons to evaluate		

baselinewma (/host/key,data period:time shift,season_unit,num_seasons)

Calculates the baseline by averaging data from the same time-frame in multiple equal time periods ('seasons') using the weighted moving average algorithm.

data period - the data gathering period within a season, defined as <N><time unit> where N - number of time units time unit - h (hour), d (day), w (week), M (month) or y (year), must be equal to or less than season

Time shift - the time period offset, defines the end of data gathering time frame in seasons (see examples)

season_unit - duration of one season (h, d, w, M, y), cannot be smaller than data period

num_seasons - number of seasons to evaluate

Examples: => **base-linewma**(/host/key,1h:n
→ calculating baseline based on the last full hour within a 3-day period that ended yesterday. If "now" is Monday 13:30, the data for 12:00-12:59 on Friday, Saturday, and Sunday will be analyzed.

=> **base-linemwa**(/host/key,2h:n
→ calculating baseline based on the last two hours within a 3-day period that ended yesterday. If "now" is Monday 13:30, the data for 10:00-11:59 on Friday, Saturday, and Sunday will be analyzed.

=> **base-linewma**(/host/key,1d:n
→ calculating baseline based on the last days of previous 4 months,

FUNCTION

trendavg (/host/key,time period:time shift)

Average
of trend
values
within the
defined
time
period.

See
common
param-
eters.

Examples:
=> **trendavg**(/host/key,**1h:now**)
→
average
for the
previous
hour (e.g.
12:00-
13:00)
=> **trendavg**(/host/key,**1h:now**
1h) →
average
for two
hours ago
(11:00-
12:00)
=> **trendavg**(/host/key,**1h:now**
2h) →
average
for three
hours ago
(10:00-
11:00)
=> **trendavg**(/host/key,**1M:now**
1y) →
average
for the
previous
month a
year ago

trendcount (/host/key,time period:time shift)

	Number of successfully retrieved trend values within the defined time period.	See common parameters.	Examples: => trend-count(/host/key,1h:now) → count for the previous hour (e.g. 12:00-13:00) => trend-count(/host/key,1h:now 1h) → count for two hours ago (11:00-12:00) => trend-count(/host/key,1h:now 2h) → count for three hours ago (10:00-11:00) => trend-count(/host/key,1M:now 1y) → count for the previous month a year ago
trendmax	(/host/key,time period:time shift)		

	The maximum in trend values within the defined time period.	See common parameters.	Examples: => trend-max(/host/key,1h:now/ → maximum for the previous hour (e.g. 12:00-13:00) => trend-max(/host/key,1h:now/ - trend-min(/host/key,1h:now/h → calculate the difference between the maximum and minimum values (trend delta) for the previous hour (12:00-13:00) => trend-max(/host/key,1h:now/1h) → maximum for two hours ago (11:00-12:00) => trend-max(/host/key,1h:now/2h) → maximum for three hours ago (10:00-11:00) => trend-max(/host/key,1M:now,1y) → maximum for the previous month a year ago
trendmin	(/host/key,time period:time shift)		

	The minimum in trend values within the defined time period.	See common parameters.	Examples: => trend-min (/host/key, 1h:now/h → minimum for the previous hour (e.g. 12:00-13:00) => trend-min (/host/key, 1h:now/h - trend-min (/host/key, 1h:now/h → calculate the difference between the maximum and minimum values (trend delta) for the previous hour (12:00-13:00) => trend-min (/host/key, 1h:now/h 1h) → minimum for two hours ago (11:00-12:00) => trend-min (/host/key, 1h:now/h 2h) → minimum for three hours ago (10:00-11:00) => trend-min (/host/key, 1M:now/ 1y) → minimum for the previous month a year ago
--	---	-------------------------------	---

trendstl (/host/key,eval period:time shift,detection period,season,<deviations>,<devalg>,<s_window>)

Returns the rate of anomalies during the detection period - a decimal value between 0 and 1 that is ((the number of anomaly values)/(total number of values)).	eval period - the time period that must be decomposed (minimum '1h'), defined as <N><time unit> where N - number of time units time unit - h (hour), d (day), w (week), M (month) or y (year). Time shift - the time period offset (see examples) detection period - the time period before the end of eval period for which anomalies are calculated (minimum '1h', cannot be longer than eval period), defined as <N><time unit> where N - number of time units time unit - h (hour), d (day), w (week). season - the	Examples: => trend-stl (/host/key, 100h:now → analyse the last 100 hours of trend data, find the anomaly rate for the last 10 hours of that period, expecting the periodicity to be 2h, the remainder series values of the evaluation period are considered anomalies if they reach the value of 3 deviations of the MAD of that remainder series => trend-stl (/host/key, 100h:now 10h,100h,2h,2.1,"mad") → analyse the period of 100 hours of trend data, up to 10 hours ago, find the anomaly rate for that entire period expecting the periodicity to be 2h, the remainder series
--	---	---

FUNCTION

trendsum (/host/key,time period:time shift)

Sum of trend values within the defined time period.

See **common parameters**.

Examples:
=>
trendsum(/host/key,**1h:now/**
→ sum for the previous hour (e.g. 12:00-13:00)
=>
trendsum(/host/key,**1h:now/1h**) → sum for two hours ago (11:00-12:00)
=>
trendsum(/host/key,**1h:now/2h**) → sum for three hours ago (10:00-11:00)
=>
trendsum(/host/key,**1M:now/1y**) → sum for the previous month a year ago

5 Funções matemáticas

Todas as funções listadas aqui são suportadas em:

- **Expressões de gatilho**
- **Itens calculados**

Funções matemáticas são suportadas com tipos de valor float e integer, a menos que seja afirmado o contrário.

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Expressões são aceitas como parâmetros
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >

FUNÇÃO

Descrição	Parâmetros específicos da função	Comentários
-----------	----------------------------------	-------------

abs (valor)

FUNÇÃO

O valor
absoluto
de um
valor.

valor -
valor a ser
verificado

Tipos de
valor su-
portados:
float, int,
str, text,
log

Para
strings
retorna:
0 - valores
são iguais
1 - os
valores
diferem

Exemplo:
=>

abs(last(/host/key))>10

A
diferença
numérica
absoluta
será
calculada
, como
visto com
esses
valores de
exemplo
recebidos
(valor
'anterior'
e 'mais
recente'
=
diferença
absoluta):
'1' e '5' =
4
'3' e '1' =
2
'0' e '-2.5'
= 2.5

acos (valor)

FUNÇÃO

	O arco-seno de um valor como um ângulo, expresso em radianos.	valor - valor a ser verificado	O valor deve estar entre -1 e 1. Por exemplo, o arco-seno de um o valor '0.5' será '2.0943951'. Exemplo: => acos (last(/host/key))
asin (valor)	O arco-seno de um valor como um ângulo, expresso em radianos.	valor - valor a ser verificado	O valor deve estar entre -1 e 1. Por exemplo, o arco-seno de um o valor '0,5' será '-0,523598776'. Exemplo: => asin (last(/host/key))
atan (valor)	O arco tangente de um valor como um ângulo, expresso em radianos.	valor - valor a ser verificado	Por exemplo, o arco tangente de um valor '1' será '0,785398163'. Exemplo: => atan (last(/host/key))
atan2 (valor,abscissa)			

FUNÇÃO

	O arco tangente das coordenadas de ordenada (exprue) e abcissa especificadas como um ângulo, expresso em radianos.	valor - valor a ser verificado abscissa - valor de abcissa	Por exemplo, o arco tangente das coordenadas de ordenada e abcissa de um valor '1' será '2.21429744'. Exemplo: => atan2 (last(/host/key),2)
média (<valor1>,<valor2>,...)	Valor médio dos valores do item referenciado.	valueX - valor retornado por uma das funções de histórico	Exemplo: => avg (avg(/host/key),avg(host2/chave2))
cbirt (valor)	Raiz cúbica de um valor.	valor - valor a ser verificado	Por exemplo, a raiz cúbica de '64' será '4', de '63' será '3.97905721'. Exemplo: => cbirt (last(/host/key))
ceil (valor)	Arredonde o valor para o inteiro maior mais próximo.	valor - valor a ser verificado	Por exemplo, '2,4' será arredondado para '3'. Exemplo: => ceil (last(/host/key)) Veja também floor().
cos (valor)			

FUNÇÃO

cosh (valor)	O cosseno de um valor, em que o valor é um ângulo expresso em radianos.	valor - valor a ser verificado	Por exemplo, o cosseno de um valor '1' será '0,54030230586'. Exemplo: => cos (last(/host/key))
berço (valor)	O cosseno hiperbólico de um valor.	valor - valor a ser verificado	Por exemplo, o cosseno hiperbólico de um valor '1' será '1,54308063482'. Retorna o valor como real número, não como notação científica. Exemplo: => cosh (last(/host/key))
graus (valor)	A cotangente de um valor, onde o valor é um ângulo, expresso em radianos.	valor - valor a ser verificado	Por exemplo, a cotangente de um valor '1' será '0,54030230586'. Exemplo: => berço (last(/host/key))
e	Converte um valor de radianos para graus.	valor - valor a ser verificado	Por exemplo, um valor '1' convertido em graus será '57.2957795'. Exemplo: => graus (last(/host/key))
exp (valor)	Número de Euler (2.718281828459045).		Exemplo: => e ()

FUNÇÃO			
expm1 (valor)	Número de Euler em uma potência de um valor.	valor - valor a verificar	Por exemplo, o número de Euler em uma potência de um valor '2' será '7.38905609893065'. Exemplo : => exp (last(/host/key))
	Número de Euler em uma potência de um valor menos 1.	valor - valor a verificar	Por exemplo, o número de Euler em uma potência de um valor '2' menos 1 será '6,38905609893065'. Exemplo: => expm1 (last(/host/key))
piso (valor)	Arredonde o valor para o inteiro menor mais próximo.	valor - valor a ser verificado	Por exemplo, '2,6' será arredondado para '2'. Exemplo: => floor (last(/host/key))
log (valor)	Logaritmo natural.	valor - valor a ser verificado	Veja também ceil(). Por exemplo, o logaritmo natural de um valor '2' será '0,69314718055994529'
log10 (valor)			Exemplo: => ** log**(last(/host/chave))

FUNÇÃO

	Logaritmo decimal.	valor - valor a ser verificado	Por exemplo, o logaritmo decimal de um valor '5' será '0,69897000433'. Exemplo: => log10 (last(/host/chave))
max (<valor1>,<valor2>,...)	Valor mais alto dos valores de item referenciados.	valueX - valor retornado por uma das funções de histórico	Exemplo: => max (avg(/host/key),avg(host2/chave2))
min (<valor1>,<valor2>,...)	Valor mais baixo dos valores de item referenciados.	valueX - valor retornado por uma das funções de histórico	Exemplo: => min (avg(/host/key),avg(host2/chave2))
mod (valor,denominador)	Resto da divisão.	valor - valor a verificar denominador - denominador da divisão	Por exemplo, o resto da divisão de um valor '5' com denominador de divisão '2' será '1'. Exemplo: => mod (last(/host/key),2)
pi	Constante Pi (3.14159265358979).		Exemplo: => pi ()
potência (valor,valor de potência)	A potência de um valor.	valor - valor a ser verificado valor da potência - a enésima potência a ser usada	Por exemplo, a terceira potência de um valor '2' será '8'. Exemplo: => power (last(/host/key),3)

FUNÇÃO

radianos (valor)

Converter um valor de graus para radianos.

valor - valor a ser verificado

Por exemplo, um valor '1' convertido em radianos será '0,0174532925'.

Exemplo:
=> **radi-
anos**(last(/host/chave))

rand

Retorna um valor inteiro aleatório.

Um número gerado pseudoaleatoriamente usando o tempo como semente (suficiente para fins matemáticos, mas não para criptografia).

Exemplo:
=>
****rand***
*****()

round (valor, casas decimais)

Arredonde o valor para casas decimais.

valor - valor a verificar
casas decimais - especifique casas decimais para arredondamento (0 também é possível)

Por exemplo, um valor '2,5482' arredondado para 2 casas decimais será '2,55'.

Exemplo:
=>
round(last(/host/key),2)

signum (valor)

FUNÇÃO

	Retorna '-1' se um valor for negativo, '0' se um valor for zero, '1' se um valor for positivo.	valor - valor a ser verificado	Exemplo: => signum (last(/host/key))
sen (valor)	O seno de um valor, onde o valor é um ângulo expresso em radianos.	valor - valor a ser verificado	Por exemplo, o seno de um valor '1' será '0,8414709848'. Exemplo: => sin (last(/host/key))
sinh (valor)	O seno hiper- bólico de um valor.	valor - valor a ser verificado	Por exemplo, o seno hiper- bólico de um valor '1' será '1,17520119364'. Exemplo: => sinh (last(/host/chave))
sqrt (valor)	Raiz quadrada de um valor.	valor - valor a ser verificado	Esta função falhará com um valor negativo. Por exemplo, a raiz quadrada de um valor '3,5' será ' 1.87082869339'. Exemplo: => sqrt (last(/host/key))
soma (<valor1>,<valor2>,...)			

FUNÇÃO			
	Soma dos valores do item referenciado.	valorX - valor retornado por uma das funções de histórico	Exemplo: => soma (avg(/host/key),avg(/chave2))
tan (valor)	A tangente de um valor.	valor - valor a ser verificado	Por exemplo, a tangente de um valor '1' será '1,55740772465'. Exemplo: => tan (last(/host/key))
truncar (valor, casas decimais)	Truncar o valor para casas decimais.	valor - valor a verificar casas decimais - especificar casas decimais para truncar (0 também é possível)	Exemplo: => * *truncar *(last(/host/key),2)

6 Funções do operador

Todas as funções listadas aqui são suportadas em:

- [Expressões de gatilho](#)
- [Itens calculados](#)

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Expressões são aceitas como parâmetros

FUNÇÃO		
	Descrição	Parâmetros Comentários específicos da função
entre (valor,min,max)		

FUNÇÃO

Verifique se um valor pertence ao intervalo fornecido.

valor - valor a ser verificado
min - valor mínimo
max - valor máximo

Tipos de valor suportados: inteiro, flutuante
 Retorna:
 1 - no intervalo
 0 - caso contrário

Exemplo:
 => **entre**(last(/host/key),**1,10**)=1
 - dispara se o valor estiver entre 1 e 10.

in (valor,valor1,valor2,...valorN)

Verifique se um valor é igual a pelo menos um dos valores listados.	valor - valor a ser verificado valor1,valor2,...,valorN - valores listados (valores de string devem ser aspas duplas)	Tipos de valor suportados: valor1,valor2,...,valorN Retorna: 1 - se igual 0 - caso contrário O valor é comparado aos valores listados como números, se todos esses valores podem ser convertidos em numéricos; caso contrário, comparado como strings. Exemplo: => in(last(/host/key),5,10)= - acionar se o último valor é igual a 5 ou 10 => in("text",last(/host/key),last(/host, - disparar se "texto" for igual a qualquer um dos últimos 2 valores.
---	--	---

7 Funções de previsão

Todas as funções listadas aqui são suportadas em:

- Expressões de gatilho
- Itens calculados

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >
- Os parâmetros específicos da função são descritos com cada função
- Os parâmetros /host/key e (sec|#num)<:time shift> nunca devem ser citado

Parâmetros comuns

- /host/key é um primeiro parâmetro obrigatório comum para as funções referenciando o histórico de itens do host
- (sec|#num)<:time shift> é um segundo parâmetro comum para o funções que fazem referência ao histórico do item do host, onde:
 - **sec** - máximo **avaliação period** em segundos (tempo **sufixos** podem ser usados), ou
 - **#num** - máximo **avaliação range** na última valores coletados (se precedidos por uma marca de hash)
 - **time shift** (opcional) permite mover o ponto de avaliação volta no tempo. Ver mais detalhes](/manual/config/triggers/expression# em especificando a mudança de horário.

Funções de previsão

FUNÇÃO

Descrição	Parâmetros	Comentários
	específi-	
	cos da	
	função	

previsão (/host/key,(sec|#num)<:time shift>,time,<fit>,<mode>)

Valor futuro, max, min, delta ou avg do item.	Consulte parâmetros comuns.	Tipos de valor suportados: float, int
	tempo - horizonte de previsão em segundos (os sufixos de tempo podem ser usado); valores negativos são suportados	Se o valor a ser retornado for maior que 1,7976931348623157E+ ou menos que - 1,7976931348623157E+ o valor de retorno é cortado para 1,7976931348623157E+308 ou - 1,7976931348623157E+308
	fit (opcional; deve estar entre aspas duplas) - função usada para ajustar dados históricos	correspondentemente.
	compatível com fits : linear - linear função polinômioN - polinômio de grau N (1 <= N <= 6)	Torna-se incompatível apenas se usado incorretamente na expressão (tipo de item errado, parâmetros inválidos), caso contrário retorna -1 em caso de erros.
	exponencial - função exponencial logarítmico - função logarítmica potência - função de potência >	Exemplos: => previsão(/host/key,#10,1h) → valor do item previsto em uma hora com base nos últimos 10 valores => ** previsão(/host/key,1h,30m) → valor do item previsto em 30 minutos com
	Observe que: linear é o padrão, polino-	

FUNÇÃO

timeleft (/host/key,(sec|#num)<:time shift>,threshold,<fit>)

Tempo em segundos necessário para um item atingir um limite específico.	Consulte parâmetros comuns . limiar - valor a ser alcançado (sufixos de unidade podem ser usados) fit (opcional; deve ser entre aspas duplas) - veja <code>forecast()</code>	<p>Tipos de valor suportados: float, int</p> <p>Se o valor a ser retornado for maior que 1,7976931348623157E+308 o valor de retorno será recortado para 1,7976931348623157E+308</p> <p>Retorna 1,7976931348623157E+308 se o limite não puder ser alcançado.</p> <p>Torna-se incompatível somente se usado incorretamente. na expressão (tipo de item errado, parâmetros inválidos), caso contrário retorna -1 em caso de erros.</p> <p>Exemplos: => timeleft(/host/key,#10,0) → tempo até que o valor do item chegue a zero com base nos últimos 10 valores => timeleft(/host/key,1h,1) → tempo até que o valor do item</p>
---	---	---

8 funções de string

Todas as funções listadas aqui são suportadas em:

- Expressões de gatilho
- Itens calculados

Algumas notas gerais sobre parâmetros de função:

- Os parâmetros da função são separados por uma vírgula
- Expressões são aceitas como parâmetros
- Parâmetros de string devem estar entre aspas duplas; caso contrário, eles podem obter mal interpretado
- Parâmetros de função opcionais (ou partes de parâmetros) são indicados por < >

FUNÇÃO

	Descrição	Parâmetros específicos da função	Comentários
ascii (valor)	O código ASCII do caractere mais à esquerda do valor.	valor - valor a ser verificado	<p>Tipos de valor suportados: string, texto, log</p> <p>Por exemplo, um valor como 'Abc' return '65' (código ASCII para 'A').</p> <p>Exemplo: => ascii(last(/host/key))</p>
comprimento de bits (valor)	O comprimento do valor em bits.	valor - valor a verificar	<p>Tipos de valor suportados: string, texto, log, inteiro</p> <p>Exemplo: => comprimento de bits (last(/host/chave))</p>
bytlength (valor)			

FUNÇÃO

char (valor)

O comprimento do valor em bytes.

valor - valor a ser verificado

Tipos de valor suportados: string, texto, log, inteiro

Exemplo:
=> **byte-length**
(last(/host/chave))

Retorne o caractere interpretando o valor como código ASCII.

valor - valor a ser verificado

Tipos de valor suportados: integer

O valor deve estar no intervalo de 0 a 255. Por exemplo, um valor como '65' (interpretado como código ASCII) retornará 'A'.

Exemplo:
=>
char(last(/host/key))

concat (<valor1>,<valor2>,...)

A string resultante da concatenação de valores de itens referenciados ou valores constantes.	valor - um valor retornado por uma das funções de histórico ou um valor constante (string, integer ou float number)	<p>Tipos de valor suportados: string, text, log, float, integer</p> <p>Para Por exemplo, um valor como 'Zab' concatenado a 'bix' (a string constante) retornará 'Zabbix'.</p> <p>Deve conter pelo menos dois parâmetros.</p> <p>Exemplos: => concat(last(/host/key),"bix' => concat("1 min: ",last(/host/system.cpu.l 15 min: ",last(/host/system.cpu.l</p>
inserir (valor,início,comprimento,substituição)		

Insira caracteres ou espaços especificados na string de caracteres começando na posição especificada na string.	valor - valor a ser verificado start - posição inicial comprimento - posições para substituir substituição - substituição string	Tipos de valor suportados: string, texto, log exemplo, um valor como 'Zabbix' será substituído por 'Zabbix' se 'bb' (posição inicial 3, posições para substituir 2) é substituído por 'b'. Exemplo: => insert(last(/host/key),3,2,"b")
---	---	--

esquerda (valor,contagem)

FUNÇÃO

	Os caracteres mais à esquerda do valor.	valor - valor a ser verificado count - número de caracteres a serem retornados	<p>Tipos de valor suportados: string, texto, log</p> <p>Por exemplo, você pode retornar 'Zab' de 'Zabbix' especificando 3 caracteres mais à esquerda para retornar.</p> <p>Exemplo: => left(last(/host/key),**3**) - retorna três caracteres mais à esquerda</p> <p>Veja também à direita().</p>
comprimento (valor)			

FUNÇÃO

O comprimento do valor em caracteres.

valor - valor a ser verificado

Tipos de valor suportados: str, text, log

Exemplo:

=>

length(last(/host/key))

→ comprimento do valor mais recente

=>

comprimento(last(/host/key,#3))

→ comprimento do terceiro valor mais recente

=> *

length(last(/host/key,1d)) →

comprimento do valor mais recente

um dia atrás

ltrim (valor,<chars>)

	Remova os caracteres especificados do início da string.	value - valor a ser verificado chars - (opcional) especifique os caracteres a serem removidos O espaço em branco é cortado à esquerda por default (se nenhum caractere opcional for especificado).	Tipos de valor suportados: string, text, log Exemplo: => ltrim (last(/host/key)) - remove whitespaces do início da string => ltrim (last(/host/key),"Z") - remova qualquer 'Z' do início da string => ** ltrim (last(/host/key),"Z"**) - remove qualquer espaço e 'Z' do início da string Veja também: rtrim(), trim()
meio (valor,início,comprimento)			

	Retorna uma substring de N caracteres começando na posição do caractere especificada por 'start'.	value - valor a ser verificado start - posição inicial da substring **length* - posições para retornar em substring	Tipos de valor suportados: string, texto, log Por exemplo, é possível retornar 'abbi' de um valor como 'Zabbix' se a posição inicial for 2 e posições a serem retornadas é 4). Exemplo: => mid (last(/host/key),2,4)
repetir (valor,contagem)	Repita uma string.	valor - valor a verificar count - número de vezes a repetir	Tipos de valor suportados: string, texto, log Exemplo: => repetir (last(/host/key),2) - repita o valor duas vezes
substituir (valor, padrão, substituição)	Encontrar patern no valor e substituir por substituição. Todas as ocorrências do padrão serão substituídas.	valor - valor a ser verificado padrão - padrão a ser encontrado substituição - string para substituir o padrão por	Tipos de valor suportados : string, texto, log Exemplo: => replace (last(/host/key),"ibb" - substitua todos os 'ibb' por 'abb'
direita (valor,contagem)			

FUNÇÃO

	Os caracteres mais à direita do valor.	valor - valor a ser verificado count - número de caracteres a serem retornados	Tipos de valor suportados: string, texto, log Por exemplo, você pode retornar 'bix' de 'Zabbix' especificando 3 caracteres mais à direita para retornar. Exemplo: => right (last(/host/key),**3**) - retorna três caracteres mais à direita Veja também left().
rtrim (valor,<chars>)			

Remove os caracteres especificados do final da string.	value - valor a ser verificado chars - (opcional) especifique os caracteres a serem removidos O espaço em branco é cortado à direita por default (se nenhum caractere opcional for especificado).	Tipos de valor suportados: string, text, log Exemplo: => rtrim (last(/host/key)) - remove whitespaces do final da string => rtrim (last(/host/key), "x") - remova qualquer 'x' do final da string => ** rtrim (last(/host/key), "**") - remove qualquer 'x' ou espaço do final da string Veja também: ltrim(), trim()
--	---	--

trim (valor,<chars>)

FUNÇÃO

Remove os caracteres especificados do início e do final da string.	value - valor a ser verificado chars - (opcional) especifique os caracteres a serem removidos Espaço em branco é cortado de ambos os lados por padrão (se nenhum caractere opcional for especificado).	Tipos de valor suportados: string, text, log Exemplo: => trim (last(/host/key)) - remova o espaço em branco do início e do final da string => trim (last(/host/key), "_") - remova '_' do início e do final de string Veja também: ltrim(), rtrim()
--	--	--

7 Macros

1 Macros suportadas

[comentário]: # (tags: macro)

Visão geral

A tabela contém uma lista completa de macros suportadas pelo Zabbix sai da caixa.

Para ver todas as macros suportadas em um local (por exemplo, em "URL do mapa"), você pode colar o nome do local na caixa de pesquisa em na parte inferior da janela do seu navegador (acessível pressionando CTRL+F) e faça uma busca por próximo.

Macro	Suportado em	Descrição
{ACTION.ID}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Comandos e notificações de descoberta → Notificações e comandos de registro automático → Notificações internas 	ID numérico da ação acionada.

Macro	Suportado em	Descrição
{ACTION.NAME}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Comandos e notificações de descoberta → Notificações e comandos de registro automático → Notificações internas 	Nome da ação acionada.
{ALERT.MESSAGE}	→ Parâmetros do script de alerta	Valor da 'mensagem padrão' da configuração da ação.
{ALERT.SENDTO}	→ Parâmetros de script de alerta	Suportado desde 3.0.0. Valor 'Enviar para' da configuração de mídia do usuário.
{ALERT.SUBJECT}	→ Parâmetros de script de alerta	Suportado desde 3.0.0. Valor de 'assunto padrão' da configuração da ação.
{DATA}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Comandos e notificações de descoberta >→ Notificações e comandos de registro automático → Notificações internas → Ação manual do evento scripts 	Suportado desde 3.0.0. Data atual em aaaa.mm.dd. formato.
{DISCOVERY.DEVICE.IPADDRESS}	→ Notificações e comandos de descoberta	Endereço IP do dispositivo descoberto.
{DISCOVERY.DEVICE.DNS}	→ Notificações e comandos de descoberta	Disponível sempre, não depende da adição do host. Nome DNS do dispositivo descoberto.
{DISCOVERY.DEVICE.STATUS}	→ Notificações e comandos de descoberta	Disponível sempre, não depende da adição do host. Status do dispositivo descoberto: pode ser UP ou DOWN.
{DISCOVERY.DEVICE.UPTIME}	→ Notificações e comandos de descoberta	Tempo desde a última alteração do status de descoberta para um determinado dispositivo, com precisão de até um segundo. Por exemplo: 1h 29m 01s. Para dispositivos com status DOWN, esse é o período de inatividade.
{DISCOVERY.RULE.NAME}	→ Notificações e comandos de descoberta	Nome da regra de descoberta que descobriu a presença ou ausência do dispositivo ou serviço.
{DISCOVERY.SERVICE.NAME}	→ Notificações e comandos de descoberta	Nome do serviço que foi descoberto. Por exemplo: HTTP.
{DISCOVERY.SERVICE.PORT}	→ Notificações e comandos de descoberta	Porta do serviço que foi descoberto. Por exemplo: 80.

Macro	Suportado em	Descrição
{DISCOVERY.SERVICE.STATUS}	Notificações e comandos de descoberta	O status do service:// descoberto pode ser UP ou DOWN. {DISCOVERY.SERVICE.UPTIME} → Notificações e comandos de descoberta Tempo desde a última alteração do status de descoberta para um determinado serviço, com precisão de até um segundo. Por exemplo: 1h 29m 01s. Para serviços com status DOWN, este é o período de inatividade. {ESC.HISTORY} → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações internas Histórico de escalonamento. Log de mensagens enviadas anteriormente. Mostra as notificações enviadas anteriormente, em qual etapa de escalonamento elas foram enviadas e seu status (enviado//, em andamento ou falhou). Reconhecimento status do evento (Sim/Não).
{EVENT.ACK.STATUS}	→ Notificações e comandos baseados em gatilho → Notificações e comandos de atualização de problemas → Ação de evento manual scripts	
{EVENT.AGE}	→ Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações e comandos de descoberta → Notificações e comandos de registro automático → Notificações internas → Ação de evento manual scripts	Age of the evento que acionou uma ação, com precisão de até um segundo. Útil em mensagens escaladas.
{EVENT.DATE}	→ Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificação de descoberta cations e comandos → Notificações e comandos de registro automático → Notificações internas → Ação manual do evento scripts	Data do evento que acionou uma ação.
{EVENT.DURATION}	→ Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações internas → Ação manual do evento scripts	Duração do evento (diferença de tempo entre os eventos de problema e de recuperação), com precisão de até um segundo. Útil em mensagens de recuperação de problemas. Suportado desde 5.0.0.

Macro	Suportado em	Descrição
{EVENT.ID}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Notificações e comandos de atualização de problemas → Comandos e notificações baseados em serviço → Notificações e comandos de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações e comandos de descoberta → Notificações e comandos de registro automático → Notificações internas → URLs de acionamento → Ação de evento manual scripts 	ID numérico do evento que acionou uma ação.
{EVENT.NAME}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações internas → Ação de evento manual scripts 	<p>Nome do evento de problema que acionou uma ação.</p> <p>Suportado desde 4.0.0 .</p>
{EVENT.NSEVERITY}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações internas → Ação de evento manual scripts 	<p>Valor numérico da gravidade do evento.</p> <p>Valores possíveis: 0 - Não classificado, 1 - Informação, 2 - Aviso, 3 - Média, 4 - Alta, 5 - Desastre.</p> <p>Suportado desde 4.0.0.</p>
{EVENT.OBJECT}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Comandos e notificações baseados em serviço → Comandos e notificações de atualização de serviço → Notificações e comandos de recuperação de serviço comandos → Notificações e comandos de descoberta → Notificações e comandos de registro automático → Notificações internas → Ação de evento manual scripts 	<p>Valor numérico de do objeto de evento.</p> <p>Valores possíveis: 0 - Acionador, 1 - Host descoberto, 2 - Serviço descoberto, 3 - Registro automático, 4 - Item, 5 - Regra de descoberta de baixo nível.</p> <p>Suportado desde 4.4.0.</p>
{EVENT.OPDATA}	<ul style="list-style-type: none"> → Notificações e comandos baseados em gatilho → Comandos e notificações de atualização de problemas → Ação de evento manual scripts 	<p>Dados operacionais de o gatilho subjacente de um problema.</p> <p>Suportado desde 4.4.0.</p>
{EVENT.RECOVERY.DATE}	<ul style="list-style-type: none"> → Problema notificações de recuperação e comandos → Notificações e comandos de atualização de problema (se a recuperação ocorreu) → Recuperação de serviço notificações e comandos → Ação de evento manual scripts (se a recuperação ocorreu) 	Data do evento de recuperação.

Macro	Suportado em	Descrição
{EVENT.RECOVERY.ID}	→ Problema notificações de recuperação e comandos → Notificações e comandos de atualização de problemas (se a recuperação ocorreu) → Recuperação de serviço notificações e comandos → Ação de evento manual scripts (se a recuperação ocorreu)	ID numérico do evento de recuperação.
{EVENT.RECOVERY.NAME}	Problema notificações de recuperação e comandos → Notificações e comandos de atualização de problema (se a recuperação ocorreu) → Recuperação de serviço notificações e comandos → Ação de evento manual scripts (se a recuperação ocorreu)	Nome do evento de recuperação. Suportado desde 4.4.1.
{EVENT.RECOVERY.STATUS}	Problema notificações de recuperação e comandos → Notificações e comandos de atualização de problema (se a recuperação ocorreu) → Recuperação de serviço notificações e comandos → Ação manual do evento scripts (se a recuperação ocorreu)	Valor verbal do evento de recuperação.
{EVENT.RECOVERY.TAGS}	Problema [notificações de recuperação]/(manual/config	

Notas de rodapé

¹ As macros são suportadas em rótulos de mapas desde o Zabbix 1.8.

² Desde o Zabbix 2.0.3, as macros HOST.* são suportadas em parâmetros de chave de item e só funcionam em tipos de item com interfaces compatíveis, p.e. elas não funcionarão para itens do tipo "Zabbix Agent (ativo)", "Calculado", etc.

³ Dependendo do contexto, a macro será expandida de forma diferente. No Zabbix **2.0.1** ou anterior, no contexto de execução de um comando remoto, scripts da GUI, parâmetros de chave de item, interfaces (campos IP/DNS), apenas a interface principal do agente seria considerada como origem da informação. Desde a versão **2.0.2** (para cenários web desde a **2.2.0**) a macro será utilizada na interface principal do agente, entretanto, se ela estiver ausente, a interface principal de SNMP será utilizada. Se a interface SNMP também não estiver presente, a interface JMX principal será utilizada. Se a interface JMX não estiver presente, a interface principal IPMI. Em parâmetros de chave de item a interface que estiver selecionada para o item será utilizada, desde o Zabbix 2.0.3.

⁴ Apenas as funções **avg**, **last**, **max** e **min**, com segundos como parâmetros são suportadas nesta macro nos rótulos de mapas.

⁵ Suportado desde o Zabbix 2.0.3.

⁶ Suportado desde o Zabbix Zabbix 2.2.0. As macros {HOST.*} e as macros de usuário {\$MACRO} são suportadas nos campos do cenários web: Nome, Variáveis, Cabeçalhos, Arquivo de certificado SSL, Agente, HTTP proxy, Autenticação (usuário e senha), Senha SSL e Arquivo de chave SSL e nos campos de passos de cenário: Nome, URL, Post, Cabeçalhos, Código de status requerido e Texto requerido.

⁷ Suportado desde o Zabbix Zabbix 2.2.0. Apenas as funções **avg**, **last**, **max** e **min**, em seu segundo parâmetro são suportadas com macros nos nomes de gráficos. A macro {HOST.HOST<1-9>} pode ser utilizada em outras macros. Por exemplo:

```
* {Cisco switch:ifAlias[{#SNMPINDEX}].last()}
* %{{%HOST.HOST}:ifAlias[{#SNMPINDEX}].last()}}
```

⁸ Suportado desde o Zabbix 2.4.0.

⁹ As macros simples são suportadas para construir expressões de gatilho, mas não podem ser utilizadas umas dentro das outras.

¹⁰ Suportado desde o Zabbix 3.0.0.

Suporte adicional a macros de usuários

Em adição às localizações listadas, as **macros de usuário**, desde o Zabbix 2.0, suportam várias outras localizações:

- Hosts

- Interface IP/DNS
- Interface port
- Proxy Passivo
 - Porta de interface
- Itens
 - SNMPv3 - nome de contexto
 - SNMPv3 - nome de segurança
 - SNMPv3 - senha de autenticação
 - SNMPv3 - senha privada
 - SNMPv1/v2 comunidade
 - SNMP OID
 - SNMP porta
 - SSH usuário
 - SSH chave pública
 - SSH chave privada
 - SSH senha
 - Telnet usuário
 - Telnet senha
 - Calculados em sua fórmula
 - Trapper no campo "Hosts permtidos" (desde o Zabbix 2.2)
- Descoberta
 - * SNMPv3 - Nome de contexto
 - * SNMPv3 - Nome de segurança
 - * SNMPv3 - Senha de autorização
 - * SNMPv3 - Senha privada
 - * SNMPv1/v2 comunidade
 - * SNMP OID

Macros utilizadas no processo de descoberta de baixo-nível (LLD)

Temos aqui mais um tipo de macro utilizada no processo de LLD - **{#MACRO}**. Este tipo de macro é utilizada nas regras de descoberta e pode retornar, por exemplo, os nomes reais dos sistemas de arquivos, interfaces de rede, árvores SNMP.

Estas macros podem ser utilizadas para criar novas entidades (item, trigger, gráficos, hosts) a partir de protótipos. Ao descobrir um novo elemento (sistema de arquivo, interface de rede, etc) estas macros são substituídas pelos nomes reais dos elementos e podem ser utilizadas na criação das novas entidades.

Na descoberta de máquinas virtuais estas macros podem ainda ser utilizadas para criar hosts e grupos de hosts através de **protótipos**.

As macros LLD podem ser utilizadas em:

- protótipos de item
 - nomes
 - parâmetros de chaves
 - unidades
 - SNMP OIDs
 - sensores IPMI
 - fórmulas de itens calculados
 - Scripts SSH e Telnet
 - Parâmetros de itens em monitoramento de banco de dados
 - Descrições (Suportado desde o Zabbix 2.2.0)
- protótipos de gatilho
 - nomes
 - expressões (para se referenciar à um protótipo de gatilho e a consultas independentes)
 - URLs (Suportado desde o Zabbix 3.0.0)
 - Descrições (Suportado desde o Zabbix 2.2.0)
- protótipos de gráficos
 - nomes
- protótipos de host (Suportado desde o Zabbix 2.2.0)
 - nomes
 - nomes visíveis
 - nomes de protótipos de grupo
 - (consulte **lista completa**)

Algumas macros de LLD vêm em conjunto com o processo de LLD do Zabbix - {#FSNAME}, {#FSTYPE}, {#IFNAME}, {#SNMPINDEX}, {#SNMPVALUE}. De qualquer forma o uso destas macros não é obrigatório quando você estiver criando um LLD. Você poderá utilizar qualquer outro nome de macro e referenciá-lo.

2 User macros supported by location

Visão geral

Esta seção contém uma lista de locais, onde As macros **definíveis pelo usuário** são suportado.

Note:

Somente macros de usuário de nível global são compatíveis com Ações, Descoberta de rede, Proxies e todos os locais listados em Outros locais desta página. Nos locais mencionados, o nível de host e macros em nível de modelo não serão resolvidas.

Ações

Em **ações**, as macros do usuário podem ser utilizado nos seguintes campos:

[Localização]<[Várias macros/misturadas com texto¹ | |-----|-----| |Notificações e comandos baseados em trigger|<[yes| |Notificações internas baseadas em gatilho|<[sim| |Notificações de atualização de problemas|<[sim| |Notificações e comandos baseados em serviço|<[yes| |Notificações de atualização de serviço|<[sim| |Condição do período de tempo|<[não| |Operações|<[<| |<[Duração da etapa de operação padrão|não| |^|Duração da etapa|não]

Hosts/protótipos de host

Na configuração de um **host** e **protótipo de host**, macros de usuário podem ser usadas nos seguintes campos:

Localização		Macros múltiplas/misturado com texto ¹
IP/DNS de interface		apenas DNS
Porta de interface		no
SNMP v1, v2		
	Comunidade SNMP	yes
SNMP v3		
	Nome de contexto	yes
	Nome de segurança	yes
	Senha passe de autenticação	yes
	Senha passe de privacidade	yes
IPMI		
	Nome de usuário	yes
	Senha	yes
Etiquetas		
	Nomes de etiqueta	yes
	Valores de etiqueta	yes

Items / item prototypes

In an **item** or an **item prototype** configuration, user macros can be used in the following fields:

Location		Multiple macros/mix with text ¹
Name (deprecated)		yes
Item key parameters		yes
Update interval		no
Custom intervals		no
History storage period		no
Trend storage period		no
Description		yes
Calculated item		
	Formula	yes
Database monitor		
	Username	yes
	Password	yes
	SQL query	yes

Location		Multiple macros/mix with text ¹
HTTP agent	URL ²	yes
	Query fields	yes
	Timeout	no
	Request body	yes
	Headers (names and values)	yes
	Required status codes	yes
	HTTP proxy	yes
	HTTP authentication username	yes
	HTTP authentication password	yes
	SSI certificate file	yes
	SSI key file	yes
	SSI key password	yes
	Allowed hosts	yes
JMX agent		
	JMX endpoint	yes
Script item		
	Parameter names and values	yes
SNMP agent		
	SNMP OID	yes
SSH agent		
	Username	yes
	Public key file	yes
	Private key file	yes
	Password	yes
TELNET agent	Script	yes
	Username	yes
	Password	yes
	Script	yes
Zabbix trapper		
	Allowed hosts	yes
Tags		
	Tag names	yes
	Tag values	yes
Preprocessing		
	Step parameters (including custom scripts)	yes

Low-level discovery

In a **low-level discovery rule**, user macros can be used in the following fields:

Location		Multiple macros/mix with text ¹
Name		yes
Key parameters		yes
Update interval		no
Custom interval		no
Keep lost resources period		no
Description		yes
SNMP agent		
	SNMP OID	yes
SSH agent		
	Username	yes
	Public key file	yes
	Private key file	yes
	Password	yes
	Script	yes
TELNET agent		
	Username	yes
	Password	yes

Location		Multiple macros/mix with text ¹
Zabbix trapper	Script	yes
	Allowed hosts	yes
Database monitor	Additional parameters	yes
JMX agent	JMX endpoint	yes
HTTP agent	URL ²	yes
	Query fields	yes
	Timeout	no
	Request body	yes
	Headers (names and values)	yes
	Required status codes	yes
	HTTP authentication username	yes
	HTTP authentication password	yes
Filters	Regular expression	yes
Overrides	Filters: regular expression	yes
	Operations: update interval (for item prototypes)	no
	Operations: history storage period (for item prototypes)	no
	Operations: trend storage period (for item prototypes)	no

Network discovery

In a **network discovery rule**, user macros can be used in the following fields:

Location		Multiple macros/mix with text ¹
Update interval	SNMP v1, v2	no
	SNMP community	yes
SNMP v3	SNMP OID	yes
	Context name	yes
	Security name	yes
	Authentication passphrase	yes
	Privacy passphrase	yes
	SNMP OID	yes

Proxies

In a **proxy** configuration, user macros can be used in the following field:

Location	Multiple macros/mix with text ¹
Interface port (for passive proxy)	no

Templates

In a **template** configuration, user macros can be used in the following fields:

Location		Multiple macros/mix with text ¹
Tags	Tag names	yes
	Tag values	yes

Triggers

In a **trigger** configuration, user macros can be used in the following fields:

Location	Multiple macros/mix with text ¹
Name	yes
Operational data	yes
Expression (only in constants and function parameters; secret macros are not supported).	yes
Description	yes
URL ²	yes
Tag for matching	yes
Tags	Tag yes names Tag yes val- ues

Web scenario

In a **web scenario** configuration, user macros can be used in the following fields:

Location		Multiple macros/mix with text ¹
Name		yes
Update interval		no
Agent		yes
HTTP proxy		yes
Variables (values only)		yes
Headers (names and values)		yes
Steps		
	Name	yes
	URL ²	yes
	Variables (values only)	yes
	Headers (names and values)	yes
	Timeout	no
	Required string	yes
	Required status codes	no
Authentication		
	User	yes
	Password	yes
	SSL certificate	yes
	SSL key file	yes
	SSL key password	yes
Tags		
	Tag names	yes
	Tag values	yes

Other locations

In addition to the locations listed here, user macros can be used in the following fields:

Location	Multiple macros/mix with text ¹
Global scripts (script, SSH, Telnet, IPMI), including confirmation text Webhooks	yes JavaScript script JavaScript script parameter name JavaScript script parameter value
Monitoring → Dashboards	URL ² yes field of dynamic URL dashboard widget
Administration → Users → Media	Wheno active
Administration → General → GUI	Working time
Administration → Media types → Message templates	Subject Message

For a complete list of all macros supported in Zabbix, see [supported macros](#).

Footnotes

¹ If multiple macros in a field or macros mixed with text are not supported for the location, a single macro has to fill the whole field.

² URLs that contain a **secret macro** will not work, as the macro in them will be resolved as "*****".

8 Símbolos de unidade

Visão geral

Ter que usar alguns números grandes, por exemplo '86400' para representar o número de segundos em um dia, é difícil e propenso a erros. Isto é por que você pode usar alguns símbolos de unidade apropriados (ou sufixos) para simplificar Expressões de gatilho do Zabbix e chaves de item.

Em vez de '86400' para o número de segundos, você pode simplesmente digitar '1d'. Os sufixos funcionam como multiplicadores.

Sufixos de tempo

Para o tempo você pode usar:

- **s** - segundos (quando usado, funciona da mesma forma que o valor bruto)
- **m** - minutos
- **h** - horas
- **d** - dias
- **w** - semanas

Os sufixos de hora suportam apenas números inteiros (portanto, '1h' é suportado, '1,5h' ou '1.5h' não são; use '90m').

Sufixos de tempo são suportados em:

- trigger **expressão** constantes e parâmetros de função
- constantes de **calculadas item** fórmulas
- parâmetros do **zabbix[queue,<from>,<to>]** **item interno**
- parâmetro de período de tempo de **agregado cálculos**
- configuração do item ('Intervalo de atualização', 'Intervalos personalizados', 'Histórico campos de período de armazenamento' e 'Período de armazenamento de tendência')
- configuração do protótipo do item ('Intervalo de atualização', 'Intervalos personalizados', campos 'Período de armazenamento do histórico' e 'Período de armazenamento de tendências')
- configuração de regra de descoberta de baixo nível ('Intervalo de atualização', 'Personalizar intervalos', campos 'Manter recursos perdidos')
- configuração de descoberta de rede (campo 'Intervalo de atualização')
- configuração do cenário web (campos 'Intervalo de atualização', 'Tempo limite')
- configuração de operação de ação ('Duração da etapa de operação padrão', campos 'Duração da etapa')
- configurações do perfil do usuário ('Logout automático', 'Atualizar', 'Tempo limite da mensagem' Campos)
- gráfico **widget** de Monitoramento → Painel (campo 'Mudança de horário')
- Administração → Geral → Arrumação (período de armazenamento Campos)
- Administração → Geral → Ativar opções de exibição ('Exibir gatilhos OK para', 'Ativar gatilhos de mudança de status piscam para' Campos)
- Administração → Geral → Outro ('Intervalo de bloqueio de login' campo e campos relacionados à comunicação com o servidor Zabbix)
- Servidor Zabbix **ha_set_failover_delay=delay tempo de execução control** opção

Sufixos de memória

Os sufixos de tamanho de memória são suportados em:

- **expressão de trigger** e parâmetros de função
- formulas e constantes de **item calculado**

Para o tamanho da memória, você pode usar:

- **K** - kilobyte
- **M** - megabyte
- **G** - gigabyte
- **T** - terabyte

Outros usos

Símbolos de unidade também são usados para uma representação de dados legível por humanos no front-end.

Tanto no servidor Zabbix quanto no frontend, esses símbolos são suportados:

- **K** - quilo
- **M** - mega
- **G** - giga
- **T** - tera

Quando os valores do item em B, Bps são exibidos no frontend, a base 2 é aplicado (1K = 1024). Caso contrário, uma base de 10 é usada (1K = 1000).

Além disso, o frontend também suporta a exibição de:

- **P** - peta
- **E** - ex
- **Z** - zetta
- **Y** - yotta

Exemplos de uso

Usando alguns sufixos apropriados, você pode escrever expressões de gatilho que são mais fáceis de entender e manter, por exemplo, essas expressões:

```
last(/host/system.uptime[])<86400s
avg(/host/system.cpu.load,600s)<10
last(/host/vm.memory.size[disponível])<20971520
```

poderia ser alterado para:

```
last(/host/system.uptime[])<1d
avg(/host/system.cpu.load,10m)<10
last(/host/vm.memory.size[disponível])<20M
```

9 Sintaxe do período de tempo

Visão geral

Para definir um período de tempo, o seguinte formato deve ser usado:

`d-d, hh:mm-hh:mm`

onde os símbolos representam o seguinte:

Símbolo	Descrição
d	Dia da semana: 1 - segunda, 2 - terça ,... , 7 - domingo
hh	Horário: 00-24
mm	Minutos: 00-59

Você pode especificar mais de um período de tempo usando um ponto e vírgula (;) separador:

`d-d, hh:mm-hh:mm; d-d, hh:mm-hh:mm . . .`

Deixar o período vazio é igual a 01-07,00:00-24:00, que é o valor padrão.

::: não importante O limite superior de um período de tempo não está incluído. Assim, se você especificar 09:00-18:00 o último segundo incluído na hora período é 17:59:59. :::

1 Descrição

Símbolo	Descrição
d	Dia da semana: 1 - Segunda, 2 - Terça ,... , 7 - Domingo
hh	Horas: 00-24
mm	Minutos: 00-59

2 Padrão

Deixar um período de tempo vazio é o mesmo que definir 01-07,00:00-24:00, que é o valor padrão.

Attention:

O limite de um período de tempo não é incluído. Assim, se você definir 09:00–18:00 o último momento considerado no período de tempo será 17:59:59. É assim que é tratado desde o Zabbix 1.8.7, e sempre foi tratado desta forma nos **horários de trabalho**.

3 Exemplos

Horário de trabalho. Segunda - Sexta das 9:00 às 18:00:

1–5,09:00–18:00

Horário de trabalho com final de semana. Segunda - Sexta das 9:00 às 18:00 e sábados e domingos das 10:00 às 16:00:

1–5,09:00–18:00;6–7,10:00–16:00

10 Execução de comandos

O Zabbix usa funcionalidades comuns para verificações externas, parâmetros do usuário, itens system.run, scripts de alerta personalizados, comandos remotos e scripts globais.

Etapas de execução

O comando/script é executado de forma semelhante no Unix e no Windows plataformas:

1. Zabbix (o processo pai) cria um canal para comunicação
2. Zabbix define o pipe como saída para o filho a ser criado processar
3. Zabbix cria o processo filho (executa o comando/script)
4. Um novo grupo de processos (no Unix) ou um trabalho (no Windows) é criado para o processo filho
5. Zabbix lê do pipe até que o tempo limite ocorra ou ninguém esteja escrevendo para a outra extremidade (TODOS os identificadores/descriptores de arquivo foram fechados). Observe que o processo filho pode criar mais processos e sair antes de sair ou fechar o descriptor de identificador/arquivo.
6. Se o tempo limite não for atingido, o Zabbix espera até que o o processo filho é encerrado ou ocorre o tempo limite
7. Se o processo filho inicial foi encerrado e o tempo limite não foi alcançado, o Zabbix verifica o código de saída do processo filho inicial e compara com 0 (valor diferente de zero é considerado como falha de execução, apenas para scripts de alerta personalizados, comandos remotos e scripts de usuário executado no servidor Zabbix e proxy Zabbix)
8. Neste ponto, assume-se que tudo está feito e todo o árvore de processos (ou seja, o grupo de processos ou o trabalho) é encerrado

::: não importante O Zabbix assume que um comando/script foi feito processamento quando o processo filho inicial saiu E nenhum outro processo ainda está mantendo o identificador de saída/descriptor de arquivo aberto. Quando processamento é feito, TODOS os processos criados são finalizados. :::

Todas as aspas duplas e barras invertidas no comando são escapadas com barras invertidas e o comando é colocado entre aspas duplas.

Verificação de código de saída

O código de saída é verificado com as seguintes condições:

- Apenas para scripts de alerta personalizados, comandos remotos e scripts de usuário executado no servidor Zabbix e no proxy Zabbix.
- Qualquer código de saída diferente de 0 é considerado como execução falha.
- Conteúdo de erro padrão e saída padrão para execuções com falha são coletados e disponibilizados no frontend (onde o resultado da execução é exibido).
- Entrada de log adicional é criada para comandos remotos no servidor Zabbix para salvar a saída de execução do script e pode ser ativado usando Agente LogRemoteCommands **parâmetro**.

Possíveis mensagens de front-end e entradas de log para comandos/scripts com falha:

- Conteúdo de erro padrão e saída padrão para execuções com falha (caso existam).
- "Processo encerrado com código: N." (para saída vazia e código de saída não igual a 0).
- "Processo encerrado por sinal: N." (para processo terminado por um sinal, somente no Linux).
- "Processo finalizado inesperadamente." (para processo encerrado por razões desconhecidas).

Leia mais sobre:

- **Externo verificações**

- [Parâmetros do usuário](#)
- [system.run](#) itens
- [Scripts de alerta personalizados](#)
- [Controlo remoto comandos](#)
- [Global scripts](#)

See also

- [External checks](#)
- [User parameters](#)
- [system.run](#) itens
- [Custom alert scripts](#)
- [Remote commands](#)
- [Global scripts](#)

11 Compatibilidade de versões

Agentes suportados

Para ser compatível com a versão 6.0 do Zabbix. O agente Zabbix não pode ser mais antigo que a versão 1.4 e mais recente que a 6.0

Talvez você precise revisar a configuração de agentes mais antigos, pois alguns parâmetros mudaram, por exemplo, parâmetros relacionados a [logging](#) para versões anteriores que 3.0.

Para aproveitar ao máximo as métricas mais recentes, desempenho aprimorado e uso de memória reduzido, use o agente compatível mais recente..

Servidores Proxy suportados

É necessário que o Zabbix Proxy e o Zabbix server estejam na mesma versão. Proxies de versões anteriores não serão suportados pela versão mais atual do Zabbix.

Proxies Zabbix suportados

Para ser compatível com o Zabbix 6.0, o proxy precisa ser da mesma versão mais recente; portanto, apenas as versões de proxy Zabbix 6.0.x funcionam com servidores Zabbix 6.0.x.

::: não importante Não ficando possível iniciar a atualização do servidor e ter um antigo, o proxy desatualizado relata os dados para a versão mais atual. Essa prática, nunca foi recomendada e nem apoiada pela Zabbix, agora oficialmente desabilitado, o servidor irá ignorar todos os dados recebidos de proxies desatualizados . Veja também [upgrade procedure](#). :::

Avisos sobre o uso de versões incompatíveis do daemon Zabbix são registrados.

Supported XML files

XML files not older than version 1.8 are supported for import in Zabbix 6.0.

Attention:

In the XML export format, trigger dependencies are stored by name only. If there are several triggers with the same name (for example, having different severities and expressions) that have a dependency defined between them, it is not possible to import them. Such dependencies must be manually removed from the XML file and re-added after import.

14 Biblioteca Python para API do Zabbix

Visão geral

O [zabbix_utils](#) é uma biblioteca de Python para:

- trabalhar com a API do Zabbix;
- agir como um remetente do Zabbix;
- agir como um receptor do Zabbix .

É suportado para Zabbix 5.0, 6.0, 6.4 e versões posteriores.

14 Tratamento de erros do banco de dados

Se o Zabbix detectar que o banco de dados backend não está acessível, ele envie uma mensagem de notificação e continue as tentativas de se conectar ao base de dados. Para alguns mecanismos de banco de dados, códigos de erro específicos são reconhecido.

MySQL

- CR_CONN_HOST_ERROR
- CR_SERVER_GONE_ERROR
- CR_CONNECTION_ERROR
- CR_SERVER_LOST
- CR_UNKNOWN_HOST
- ER_SERVER_SHUTDOWN
- ER_ACCESS_DENIED_ERROR
- ER_ILLEGAL_GRANT_FOR_TABLE
- ER_TABLEACCESS_DENIED_ERROR
- ER_UNKNOWN_ERROR

14 Upgrade de monitoramento de serviço

Overview No Zabbix 6.0, a funcionalidade de [monitoramento de serviços](#) foi significativamente reformulada (veja [What's new in Zabbix 6.0.0](#) para a lista de mudanças).

Esta página descreve como os serviços e SLAs, definidos em versões anteriores do Zabbix, são alterados durante um upgrade para Zabbix 6.0 ou mais recente.

Serviços Nas versões anteriores de Zabbix, os serviços tinham dois tipos de dependência: soft e hard. Depois do upgrade, todas as dependências serão iguais. Se o "Child service" estava previamente ligado a "Parent service 1" via dependência hard e adicionalmente "Parent service 2" via dependência soft, depois do upgrade o "Child service" terá dois serviços "Parent service 1" e "Parent service 2". O mapeamento Trigger-based entre os problemas e serviços foram substituídos por mapeamento tag-based.

No Zabbix 6.0 e versões mais recentes, a configuração de serviço tem um novo parâmetro Problem tags, que permite especificar um ou vários pares de nome valor de tag para correspondência de problemas. Triggers que foram vinculados a um serviço receberão uma tag nova ServiceLink : <trigger ID>:<trigger name> (o valor da tag será truncado para 32 caracteres). Serviços vinculados irão receber ServiceLink **problem tag** com o mesmo valor.

COMENTÁRIOhttps://git.zabbix.com/projects/WEB/repos/documentation/browse/en/manual/appendix/services_upgrade.md?at=refs%2Fheads%2Frelease%2F6.0

CONTEXTO 58b995d0

Fonte: [Documentation 6.0•manual/appendix/services_upgrade.xliff](#)

Regras de cálculo de status

O 'algoritmo de cálculo de status' será atualizado usando as seguintes regras:

- Não calcular → Configure status para OK
- Problema, se pelo menos um filho tiver um problema → Mais crítico dos serviços filhos
- Problema, se todos os filhos tiverem problemas → Mais crítico se todos os filhos tiverem problemas

Note:

Se você atualizou o Zabbix pre-6.0 para o Zabbix 6.0.0, 6.0.1 ou 6.0.2, veja [Known issues](#) para documentação Zabbix 6.0.

SLAs Anteriormente, as targets SLA tinham que ser definidas para cada serviço separadamente. Desde o Zabbix 6.0, SLA se tornou uma entidade separada, que contém informações sobre o cronograma do serviço, objetivo de nível de serviço esperado (SLO) e período de inatividade a serem excluídos do cálculo. Uma vez configurado, uma SLA pode ser atribuído a vários serviços através de **service tags**.

Durante uma atualização:

- Os SLAs idênticos definidos para cada serviço serão agrupados e um SLA será criado para cada grupo.

- Cada serviço afetado receberá uma tag especial SLA:<ID> e a mesma tag será especificada no parâmetro Service tags do SLA correspondente.
- Horário de criação do serviço, uma nova métrica nos relatórios de SLA, será definido como 01/01/2000 00:00 para os serviços existentes.

15 Biblioteca de links dinâmicos do remetente Zabbix para Windows

Em um ambiente Windows, os aplicativos podem enviar dados para o Zabbix servidor/proxy diretamente usando a biblioteca de links dinâmicos do remetente Zabbix (zabbix_sender.dll) em vez de ter que iniciar um processo externo (zabbix_sender.exe).

A biblioteca de vínculo dinâmico com os arquivos de desenvolvimento está localizada em pastas bin\winXX\dev. Para usá-lo, inclua o cabeçalho zabbix_sender.h arquivo e link com a biblioteca zabbix_sender.lib. Um arquivo de exemplo com O uso da API do remetente do Zabbix pode ser encontrado em pasta build\win32\examples\zabbix_sender.

A seguinte funcionalidade é fornecida pelo remetente dinâmico do Zabbix biblioteca de links:

```
|int zabbix_sender_send_values(const char *address, unsigned short port,const char *source, const zabbix_s
|-----|-----|-----|-----|-----|-----|-----|-----|-----| |char
**resultado);{.c}<|<|
```

As seguintes estruturas de dados são usadas pelo link dinâmico do remetente do Zabbix biblioteca:

```
estrutura typedef
{
    /* nome do host, deve corresponder ao nome do host de destino no Zabbix */
    char *host;
    /* a chave do item */
    caractere *chave;
    /* valor do item */
    caractere *valor;
}
zabbix_sender_value_t;

estrutura typedef
{
    /* número de valores totais processados */
    inteiro total;
    /* número de valores com falha */
    int falhou;
    /* tempo em segundos que o servidor gastou processando os valores enviados */
    tempo_gasto duplo;
}
zabbix_sender_info_t;
```

18 Agente vs agente 2 comparação

Esta seção descreve as diferenças entre o Agente Zabbix e Agente Zabbix 2.

Parâmetro	Agente Zabbix	Agente Zabbix 2
Linguagem de programação	C	Go com algumas partes em C
Daemonization	sim	apenas através de systemd (no Windows)
Extensões suportadas	Customização de módulos carregáveis em C.	Customização de plugins em Go.
Requisitos		
Plataformas suportadas	Linux, IBM AIX, FreeBSD, NetBSD, OpenBSD, HP-UX, Mac OS X, Solaris: 9, 10, 11, Windows: todas as versões desktops e servidores desde XP.	Linux, Windows: todas as versões desktops e servidores desde XP.

Parâmetro	Agente Zabbix	Agente Zabbix 2
Bibliotecas de criptografia suportadas	GnuTLS 3.1.18 e mais recentes OpenSSL 1.0.1, 1.0.2, 1.1.0, 1.1.1 LibreSSL - testado com as versões 2.7.4, 2.8.2 (aplicam-se certas limitações, veja a página Encryption para mais detalhes).	Linux: OpenSSL 1.0.1 e posterior é suportado desde o Zabbix 4.4.8. MS Windows: OpenSSL 1.1.1 ou posterior. A biblioteca OpenSSL deve ter o suporte PSK habilitado. LibreSSL não é suportado.
Monitoramento de processos		
Processo	Um processo de verificação ativo separado para cada registro de servidor/proxy.	Processo único com threads criados automaticamente. O número máximo de threads é determinado pela variável de ambiente GOMAXPROCS.
Mettricas	UNIX: veja a lista de suportada dos itens . Windows: veja uma lista adicional de itens específicos para Windows itens .	UNIX: Todas as métricas suportadas pelo agente Zabbix. Além disso, o agente 2 fornece solução de monitoramento nativa do Zabbix para: Docker, Memcached, MySQL, PostgreSQL, Redis, systemd, e outros alvos de monitoramento - veja uma lista completa dedas especificações do agente 2 itens . Windows: Todas as métricas suportadas pelo agente Zabbix, e também net.tcp.service* verificações de HTTPS, LDAP. Além disso, o agente 2 fornece solução de monitoramento nativa do Zabbix para: PostgreSQL, Redis.
Simultaneidade	Verificações ativas para um único servidor são executadas sequencialmente.	Verificações de diferentes plugins ou múltiplas verificações dentro de um plugin podem ser executadas simultaneamente.
Intervalos programados/flexíveis	Suportado apenas para verificações passivas.	Suportado para verificações passivas e ativas.
Traps de terceiros	não	sim
Recursos adicionais		
Armazenamento persistente	não	sim
Arquivos persistentes para log*[] métricas	sim (apenas para Unix)	não
Configurações de tempo limite	Definido apenas no nível do agente .	O tempo limite do plug-in pode substituir o tempo limite definido no nível do agente.
Alterar usuário em tempo de execução	sim (somente sistemas do tipo Unix)	não (controlado por systemd)
Conjunto de criptografia configuráveis pelo usuário	sim	não

Veja também:

- Descrição dos processos Zabbix: [agente Zabbix](#), [agente Zabbix 2](#)
- Parâmetros de configuração: agente Zabbix [UNIX](#) / [Windows](#), agente Zabbix 2 [UNIX](#) / [Windows](#)

18 Exemplos de Escaping

Visão geral

Esta página fornece exemplos de como usar escaping ao utilizar expressões regulares em contextos diferentes.

Note:

Quando usar o construtor da expressão trigger, a escaping correta nas expressões regulares é adicionada automaticamente.

Exemplo

Macro do usuário com contexto

Expressão regular: `\.+\\"[a-z]+
` Macro do usuário com contexto: `{${MACRO}:regex:\".+\\\"[a-z]+\"}`

Observe que:

- barras invertidas não estão escapadas **not escaped**;
- aspas estão escapadas.

Função da macro LLD

Expressão macro: `\.+\\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\\\"[a-z]+\", \\1)}`

Observe que:

- barras investidas não estão escapadas;
- aspas estão escapadas.

Função da macro LLD dentro do contexto da macro do usuário

Expressão regular: `\.+\\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\\\"[a-z]+\", \\1)
` Macro do usuário com contexto: `{${MACRO}:\"{{#MACRO}}.iregsub(\".+\\\"[a-z]+\\\", \\1)}\"}`

Observe que:

- a barra invertida escape para LLD não muda;
- ao inserir ~~a macro LLD no contexto da macro~~ do usuário, precisamos colocá-la na string:

1. Aspas são adicionadas na expressão macro;
2. Aspas são escapadas; 3 novas barras invertidas são adicionadas no total.

Parâmetro de string de função não histórico

Conteúdo da string: `\.+\\"[a-z]+
` Expressão: `concat(\"abc\", \"\\.\\.\\.\\.\"[a-z]+\")`

Observe que:

- Parâmetros de String exigem escape para barras investidas e aspas.

Parâmetro de String de função de histórico

Conteúdo da String: `\.+\\"[a-z]+
` Expressão: `find(__ITEM_KEY__, \"regex\", \".+\\\"[a-z]+\")`

Observe que:

- barras investidas não estão escapadas;
- aspas estão escapadas.

Função do Macro LLD dentro do parâmetro de string de função não histórico

Expressão regular: `\.+\\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\\\"[a-z]+\", \\1)
` Expressão: `concat(\"abc\", \"{{#MACRO}}.iregsub(\"\\.\\.\\.\\.\"[a-z]+\\\", \\1)}\")`

Observe que:

- Os parâmetros da String exigem escape para barras investidas e aspas;
- Outra camada de escape é adicionada, porque a macro será resolvida apenas após a string ser desmarcada;

Função macro LLD dentro do parâmetro de string de função de histórico `\.+\\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\\\"[a-z]+\", \\1)
` Expressão: `find(__ITEM_KEY__, \"eq\", \"{{#MACRO}}.iregsub(\"\\.\\.\\.\\.\"[a-z]+\\\", \\1)}\")`

Observe que:

- barras investidas não estão escapadas;
- aspas estão escapadas.

Macro do usuário com contexto dentro do parâmetro de string de função não histórico

Expressão regular: `\.+\\"[a-z]+
` Macro do usuário com contexto: `{${MACRO}:regex:\".+\\\"[a-z]+\"}
` Expressão: `concat(\"abc\", \"${MACRO}:regex:\"\\.\\.\\.\\.\"[a-z]+\\\")`

Observe que:

- Assim como no exemplo anterior, uma camada extra de escape foi adicionada
- Barras invertidas e aspas são escape somente no top-level escape (em virtude de ser um parâmetro de string).

Macro do usuário com contexto dentro do parâmetro de string de função de histórico

Expressão regular: `\.+\"[a-z]+
` Macro usuário com contexto: `{${MACRO:regex:\".+\"[a-z]+}
` Expressão: `find(__ITEM_KEY__, \"eq\", \"${MACRO:regex:\".+\"[a-z]+}\")`

Observe que:

- barras invertidas não estão escapadas;
- aspas estão escapadas:

Função macro LLD dentro do macro do usuário com contexto dentro de função não histórico

Expressão regular: `\.+\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)
` Macro usuário com contexto: `{${MACRO: \"{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)}\"
` Expressão: `concat(\"abc\", \"${MACRO: \"{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)}\")`

Observe que há três camadas de escape:

1. Para a função macro LLD, sem escape de barras invertidas;
2. Para macro usuário com contexto, sem escape de barras invertidas;
3. Para o parâmetro da string de uma função, com escape de barras invertidas.

Função macro LLD dentro da macro do usuário com contexto dentro de função de histórico

Expressão regular: `\.+\"[a-z]+
` Macro LLD: `{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)
` Macro usuário com contexto: `{${MACRO: \"{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)}\"
` Expressão: `find(__ITEM_KEY__, \"eq\", \"${MACRO: \"{{#MACRO}}.iregsub(\".+\"[a-z]+\", \\1)}\")`

Observe que:

- barras invertidas não estão escapadas;
- aspas estão escapadas.

Macro do usuário com contexto apenas dentro da string

Expressão regular: `\.+\"[a-z]+
` Usuário macro com contexto: `{${MACRO:regex:\".+\"[a-z]+}
` Dentro da string de uma expressão, por exemplo: `func(arg1, arg2, arg3)=\"${MACRO:regex:\".+\"[a-z]+}\"`

Observe que:

- As Strings também exigem barras invertidas escapadas;
 - As Strings também exigem aspas escapadas;
 - Novamente um caso com 2 níveis de escape:
1. Escape para contexto macro do usuário com contexto sem barra invertida escape;
 2. Escape por ser uma string com barra invertida escape.

18 Outros problemas

Login e systemd

Recomendamos **criando** um usuário zabbix usuário como usuário do sistema, ou seja, sem capacidade de efetuar login. Alguns usuários ignore esta recomendação e use a mesma conta para fazer login (e. g.usando SSH) para hospedar o Zabbix em execução. Isso pode travar o daemon do Zabbix no log externo. Neste caso, você obterá algo como o seguinte no Zabbix registro do servidor:

```
zabbix_server [27730]: [file:'selfmon.c',line:375] lock failed: [22] Invalid argument
zabbix_server [27716]: [file:'dbconfig.c',line:5266] lock failed: [22] Invalid argument
zabbix_server [27706]: [file:'log.c',line:238] lock failed: [22] Invalid argument
```

e no log do agente Zabbix: `zabbix_agentd [27796]: [file:'log.c',line:238] lock failed: [22] Invalid argument`

Isso acontece por causa da configuração padrão do systemd `RemoveIPC=yes` configurado em `/etc/systemd/logind.conf`. Ao sair do sistema os semáforos criados pelo Zabbix anteriormente são removidos o que faz com colidam.

Uma citação da documentação do systemd:

`RemoveIPC=`

Controls whether System V and POSIX IPC objects belonging to the user shall be removed when the user fully logs out. Takes a boolean argument. If enabled, the user may not consume IPC resources after the last of the user's sessions terminated. This covers System V semaphores, shared memory and message queues, as well as POSIX shared memory and message queues. Note that IPC objects of the

root user and other system users are excluded from the effect of this setting. Defaults to "yes".

Existem 2 soluções para este problema: 1. (recommended) Stop using zabbix account for anything else than Zabbix processes, create a dedicated account for other things. 2. (não recomendado) definir RemoveIPC=no em/etc/systemd/logind.conf e reinicialize o sistema. Observe que RemoveIPC é um sistema parâmetro, alterá-lo afetará todo o sistema.

Usando o frontend Zabbix por trás do proxy

Se o frontend do Zabbix for executado atrás do servidor proxy, o caminho do cookie no arquivo de configuração do proxy precisa ser reescrito para corresponder ao caminho com proxy reverso. Veja exemplos abaixo. Se o caminho do cookie não for reescrito, os usuários podem ter problemas de autorização ao tentar entrar no frontend do Zabbix.

Configuração de exemplo para nginx

```
# ..
local / {
# ..
proxy_cookie_path /zabbix /;
proxy_pass http://192.168.0.94/zabbix/;
# ..
```

Configuração de exemplo para Apache

```
# ..
ProxyPass "/" http://host/zabbix/
ProxyPassReverse "/" http://host/zabbix/
ProxyPassReverseCookiePath /zabbix/
ProxyPassReverseCookieDomain host zabbix.example.com
# ..
```

Páginas de manual do Zabbix

Este é o manual do Zabbix para processos do Zabbix.

zabbix_agent2

Sessão: Comandos de manutenção (8)

Atualização: 2019-01-29

[Index](#) [Return to Main Contents](#)

NOME

zabbix_agent2 - Zabbix agent 2

SINOPSE

```
zabbix_agent2 [-c config-file]
zabbix_agent2 [-c config-file] -p
zabbix_agent2 [-c config-file] -t item-key
zabbix_agent2 [-c config-file] -R runtime-option
zabbix_agent2 -h
zabbix_agent2 -V
```

DESCRIÇÃO

O **zabbix_agent2** é uma aplicação para monitoramento de parâmetros de vários serviços..

OPÇÕES

-c, --config arquivo de configuração

Usa o arquivo de configuração alternativo em vez do padrão.

-R, --runtime-control opção de tempo de execução

Realiza funções administrativas de acordo com opção de tempo de execução.

Opções de controle em tempo de execução: **userparameter_reload**

Recarrega os parâmetros do usuário a partir do arquivo de configuração

aumenta loglevel

Aumenta o nível de log

diminui loglevel

Diminui o nível de log

ajuda

Lista as opções de controle em tempo de execução disponíveis

métricas

Lista as métricas disponíveis

versão

Exibe a versão

-p, --print

Imprime os itens conhecidos e sai. Para cada item, são utilizados os padrões genéricos ou padrões específicos para testes. Esses padrões são listados entre colchetes como parâmetros da chave do item. Os valores retornados são cercados por colchetes e prefixados com o tipo do valor retornado, separado por um caractere de barra vertical. Para parâmetros de usuário, o tipo é sempre **t**, pois o agente não pode determinar todos os possíveis valores de retorno. Itens exibidos como funcionando não têm garantia de funcionamento no servidor Zabbix ou com `zabbix_get` o consultar um agente em execução como `daemon`, pois as permissões ou o ambiente podem ser diferentes. Os tipos de valores retornados são:

d

Número com parte decimal.

m

Não suportado. Isso pode ser causado por consultar um item que só funciona no modo ativo, como um item de monitoramento de log ou um item que requer múltiplos valores coletados. Problemas de permissão ou parâmetros de usuário incorretos também podem resultar no estado não suportado.

s

Texto. Comprimento máximo não limitado.

t

Texto. Igual a **s**.

u

Inteiro não assinado.

-t, --test item-key

Testa um único item e sai.. Consulte **--print** para descrição da saída.

-h, --help

Exibe esta ajuda e sai.

-V, --version

Exibe informações sobre a versão e sai.

ARQUIVOS

`/usr/local/etc/zabbix_agent2.conf` Local padrão do arquivo de configuração do Zabbix agent 2 (caso não seja alterado durante a compilação dos fontes).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), **zabbix_get**(8), **zabbix_js**(8), **zabbix_proxy**(8), **zabbix_sender**(8), **zabbix_server**(8)

AUTOR

Zabbix LLC

Index

NOME

SINOPSE

DESCRIÇÃO

OPÇÕES

ARQUIVOS

VEJA TAMBÉM

AUTOR

Este documento foi criado às: 14:07:57 GMT, 22 de Novembro, 2021

zabbix_agentd

Seção: Comandos de Manutenção (8)

Atualização: 29-01-2019

[Index](#) [Retornar ao conteúdo principal](#)

NOME

zabbix_agentd - Zabbix agent daemon

SINOPSE

zabbix_agentd [-c config-file]
zabbix_agentd [-c config-file] -p
zabbix_agentd [-c config-file] -t item-key
zabbix_agentd [-c config-file] -R runtime-option
zabbix_agentd -h
zabbix_agentd -V

DESCRIÇÃO

zabbix_agentd é um daemon para monitoramento de vários parâmetros de um servidor.

OPÇÕES

-c, --config config-file

Use o config-file alternativo ao invés do padrão.

-f, --foreground

Execute o agente Zabbix em primeiro plano.

-R, --runtime-control runtime-option

Desempenhar funções administrativas de acordo com runtime-option.

Opções de controle em tempo de execução

userparameter_reload

Recarregar os parâmetros do usuário a partir do arquivo de configuração

log_level_increase[=target]

Aumenta o nível do log, afeta todos os processos se o alvo não for especificado

log_level_decrease[=target]

Diminuir o nível de log, afeta todos os processos se o alvo não for especificado

Metas de controle de nível de log

process-type Todos os processos especificados por tipo (Checagens ativas, Coletor, ouvinte)

process-type,N Tipo de processo e número(e.g., ouvinte,3)

pid Identificador do processo, até 65535. Para valores maiores especifique o destino como "process-type,N"

-p,--print

Mostra os itens conhecidos e sai. Para cada item, os padrões genéricos são usados, ou padrões específicos para teste são fornecidos. Esses padrões são listados entre colchetes como parâmetros de chave de item. Os valores retornados inseridos entre colchetes e prefixados com o tipo do valor retornado, separados pelo caractere pipe (|) . Para parâmetros do usuário o tipo é sempre **t**, pois o agente não pode determinar todos os valores de retorno possíveis. Os itens, exibidos como funcionando, não são garantidos para funcionar no servidor Zabbix ou Zabbix_get ao consultar um daemon de agente em execução. pois as permissões ou o ambiente podem ser diferentes. Os tipos de valores retornados são:

d

Número com uma parte decimal.

m Não suportado. Isso pode ser causado pela consulta de um item que funciona apenas no modo ativo, como um item de monitoramento de log ou um item que requer vários valores coletados. Problemas de permissão ou parâmetros de usuário incorretos também podem resultar no estado de não-suportado.

s Texto, Comprimento máximo não limitado.

t Texto, Igual a **s**.

u Inteiro sem sinal.

-t, --test item-key

Teste um único item e saia. Consulte **--print** para descrição de saída.

-h, --help Mostra este manual de ajuda e sai.

-V, --version Mostra as informações da versão e sai.

ARQUIVOS

/usr/local/etc/zabbix_agentd.conf

Localização padrão do arquivo de configuração do Zabbix agent (se não for modificado durante a compilação).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agent2(8), **zabbix_get**(1), **zabbix_js**(1), **zabbix_proxy**(8), **zabbix_sender**(1), **zabbix_server**(8)

AUTOR

Alexei Vladishev <alex@zabbix.com>

Index

NOME

SINOPSE

DESCRIÇÃO

OPÇÕES

ARQUIVOS

VEJA TAMBÉM

AUTOR

Esse documento foi criado em: 20:50:13 GMT, 22 de Novembro, 2021

zabbix_get

Seção: Comandos de usuários (8)

Atualizado: 6 de julho de 2021

[Índice](#) [Retornar ao conteúdo principal](#)

NOME

zabbix_get - Zabbix get utility

SINOPSE

zabbix_get -s host-name-ou-IP [-p porta] [-I Endereço-IP] [-t timeout] -k item-key

zabbix_get -s host-name-ou-IP [-p porta] [-I Endereço-IP] [-t timeout] --tls-connect cert --tls-ca-file arquivo-CA [--tls-crl-file arquivo-CRL] [--tls-agent-cert-issuer cert-issuer] [--tls-agent-cert-subject cert-subject] --tls-cert-file arquivo-cert --tls-key-file arquivo-key [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k item-key

zabbix_get -s host-name-ou-IP [-p porta] [-I Endereço-IP] [-t timeout] --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file arquivo-PSK [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k item-key

zabbix_get -h

zabbix_get -V

DESCRIÇÃO

zabbix_get é um utilitário de linha de comando para recuperar dados de um agente Zabbix.

Opções

-s, --host hostname ou IP>

Especificar hostname ou endereço IP de um host.

-p, --port <porta>

Especificar a porta de um agente rodando no host. O padrão é 10050.

-I, --source-address <endereço IP>

Especificar endereço IP de origem.

-t, --timeout segundos

Especifique o tempo limite. Faixa válida: 1-30 segundos (padrão: 30)

-k, --key item-key

Especifique a chave do item (item-key) para o qual recuperar o valor.

--tls-connect valor Como se conectar ao agente. Valores:

não criptografado

conectar sem criptografia (padrão)

psk

conectar usando TLS e uma chave pré-compartilhada

cert

conectar usando TLS e um certificado

--tls-ca-file CA-file

Caminho completo de um arquivo que contém os certificados de CA(s) de nível superior para verificação do par de certificado.

--tls-crl-file CRL-file

Caminho completo de um arquivo contendo certificados revogados.

--tls-agent-cert-issuer cert-issuer

Emissor de certificado de agente permitido

--tls-agent-cert-subject cert-subject

Assunto do certificado de agente permitido.

--tls-cert-file cert-file

Nome do caminho completo de um arquivo que contém o certificado ou cadeia de certificados.

--tls-key-file key-file

Nome do caminho completo de um arquivo que contém a chave privada.

--tls-psk-identity PSK-identity

Cadeia de identidade PSK.

--tls-psk-file PSK-file

Nome de caminho completo de um arquivo que contém a chave pré-compartilhada.

--tls-cipher13 cipher-string

Sequência de criptografia para OpenSSL 1.1.1 ou mais recente para TLS 1.3. Substituir o critérios de seleção padrão do conjunto de cifras. Esta opção não está disponível se a versão do OpenSSL é inferior a 1.1.1.

--tls-cipher cipher-string

String de prioridade GnuTLS (para TLS 1.2 e superior) ou string de cifra OpenSSL (somente para TLS 1.2). Substitua os critérios de seleção padrão do ciphersuite.

-h, --help

Exiba esta ajuda e saia.

-V, --version

Saída de informações de versão e saída.

EXEMPLOS

```
zabbix_get -s 127.0.0.1 -p 10050 -k "system.cpu.load[all,avg1]"
```

```
zabbix_get -s 127.0.0.1 -p 10050 -k "system.cpu.load[all,avg1]" --tls-connect cert --tls-ca-file /home/zabbix/zabbix_ca_file  
--tls-agent-cert-issuer "CN=Assinando CA,OU=Operações de TI,O=Exemplo Corp,DC=exemplo,DC=com" --tls-  
agent-cert-subject "CN=Servidor1,OU=Operações de TI,O=Exemplo Corp,DC=exemplo,DC=com" --tls-cert-file  
/home/zabbix/zabbix_get.crt --tls-key-file /home/zabbix/zabbix_get.key
```

```
zabbix_get -s 127.0.0.1 -p 10050 -k "system.cpu.load[all,avg1]" --tls-connect psk --tls-psk-identity "PSK ID Zabbix  
agentd" --tls-psk-file /home/zabbix/zabbix_agentd.psk
```

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), zabbix_proxy(8), zabbix_sender(1), zabbix_server(8), zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Índice

NOME

SINOPSE

DESCRIÇÃO

Opções

EXEMPLOS

VEJA TAMBÉM

AUTOR

Índice

[NAME](#)

[SINOPSE](#)

[DESCRIÇÃO](#)

[OPÇÕES](#)

[EXEMPLOS](#)

[VER TAMBÉM](#)

[AUTOR](#)

Este documento foi criado por [man2html](#), usando as páginas do manual.

Hora: 08:42:29 GMT, 11 de junho de 2021

zabbix_js

Seção: Comandos do usuário(1)

Atualizado: 2019-01-29

[Index](#) [Retornar para o Conteúdo Principal](#)

NOME

zabbix_js - Zabbix JS utility

SINOPSE

zabbix_js -s script-file **-p** input-param [**-l** log-level] [**-t** timeout]

zabbix_js -s script-file **-i** input-file [**-l** log-level] [**-t** timeout]

zabbix_js -h

zabbix_js -V

DESCRIPTION

zabbix_js é um utilitário de linha de comando que pode ser usado para testes de script incorporado.

OPÇÕES

-s, --script script-file

Especifique o nome do arquivo do script a ser executado. Se '-' for especificado como nome do arquivo, o script será lido de stdin.

-p, --param input-param

Especifique o parâmetro de entrada.

-i, --input input-file

Especifique o nome do arquivo do parâmetro de entrada. Se '-' for especificado como nome do arquivo, a entrada será lida de stdin.

-l, --loglevel log-level

Especifique o nível de log.

-t, --timeout timeout

Especifique o tempo limite em segundos. Range válido: 1-60 segundos (padrão: 10)

-h, --help

Exiba esta ajuda e saia.

-V, --version

Saída de informações de versão e saída.

EXEMPLOS

zabbix_js -s script-file.js -p example

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agent2(8), zabbix_agentd(8), zabbix_get(1), zabbix_proxy(8), zabbix_sender(1), zabbix_server(8)

Index

NOME

SINOPSE

DESCRIÇÃO

OPÇÕES

EXEMPLOS

VER TAMBÉM

Esse documento foi criado em: 21:23:35 GMT, 18 de Março, 2020

zabbix_proxy

Seção: Comandos de manutenção (8)

Atualizado: 04 de setembro de 2020

[Índice](#) [Retornar ao conteúdo principal](#)

NOME

zabbix_proxy - Zabbix proxy daemon

SINOPSE

zabbix_proxy [-c config-file]

zabbix_proxy [-c config-file] **-R** runtime-option

zabbix_proxy -h

zabbix_proxy -V

DESCRIÇÃO

zabbix_proxy é um daemon que coleta dados de monitoramento de dispositivos e os envia para o Servidor zabbix.

OPÇÕES

-c, --config config-file.

Use o config-file alternativo em vez do padrão.

-f, --foreground.

Execute o Zabbix proxy em primeiro plano

-R, --runtime-control runtime-option.

Execute funções administrativas de acordo com o runtime-option.

Opções de controle de runtime

config_cache_reload.

Recarrega o cache de configuração. O comando será ignorado se o cache estiver sendo carregado no momento. O Zabbix proxy ativo se conectará ao servidor Zabbix e solicitará os dados de configuração. Arquivo de configuração padrão (a menos que a opção **-c** seja especificada) será usado para encontrar o arquivo PID e o sinal será enviado ao processo listado no arquivo PID

snmp_cache_reload.

Recarrega o cache SNMP.

housekeeper_execute.

Executa o housekeeper. o comando será ignorado se o housekeeper já estiver em execução

diaginfo[=section].

Registra em log as informações de diagnóstico interno da seção especificada. A seção pode ser historycache, preprocessing. Por padrão todas as sessões são registradas.

log_level_increase[=target].

Aumenta o nível de detalhamento do log, afeta todos os processos se o alvo não for especificado

log_level_decrease[=target].

Diminui o nível de detalhamento do log, afeta todos os processos se o alvo não for especificado

Metas de controle de nível de registro

process-type

Todos os processos do tipo especificado (configuration syncer, data sender, discoverer, heartbeat sender, history syncer, housekeeper, http poller, icmp pinger, ipmi manager, ipmi poller, java poller, poller, self-monitoring, snmp trapper, task manager, trapper, unreachable poller, vmware collector)

process-type,N

Tipo e número do processo (por exemplo, poller,3)

pid

Identificador de processo, até 65535. Para valores maiores, especifique o destino como "tipo de processo,N"

-h, --help

Exiba esta ajuda e saia.

-V, --version

Saída de informações de versão e saída.

ARQUIVOS

/usr/local/etc/zabbix_proxy.conf Local padrão para o arquivo de configuração do proxy Zabbix (se não modificado em tempo de compilação).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), **zabbix_get**(1), **zabbix_sender**(1), **zabbix_server**(8), **zabbix_js**(1), **zabbix_agent2**(8)

AUTOR

Alexei Vladishev <alex@zabbix.com>

Índice

NOME

SINOPSE

DESCRIÇÃO

OPÇÕES

ARQUIVOS

VEJA TAMBÉM

AUTOR

Este documento foi criado em: 16:12:22 GMT, 4 de setembro de 2020

##zabbix_sender {#manpages-zabbix_sender}

Seção: Comandos do usuário (1)

Atualizado: 01/06/2021

[Índice](#) [Retornar ao conteúdo principal](#)

NOME

zabbix_sender - utilitário Zabbix sender.

SINOPSE

zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s host -k key -o value
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s host] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] -k key -o value
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s host --tls-connect cert --tls-ca-file CA-file [--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer] [--tls-server-cert-subject cert-subject] --tls-cert-file cert-file --tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k key -o value
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect cert --tls-ca-file CA-file [--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer] [--tls-server-cert-subject cert-subject] --tls-cert-file cert-file --tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect cert --tls-ca-file CA-file [--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer] [--tls-server-cert-subject cert-subject] --tls-cert-file cert-file --tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k key -o value
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect cert --tls-ca-file CA-file [--tls-crl-file CRL-file] [--tls-server-cert-issuer cert-issuer] [--tls-server-cert-subject cert-subject] --tls-cert-file cert-file --tls-key-file key-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] -s host --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k key -o value
zabbix_sender [-v] -z server [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] -k key -o value
zabbix_sender [-v] -c config-file [-z server] [-p port] [-I IP-address] [-t timeout] [-s host] --tls-connect psk --tls-psk-identity PSK-identity --tls-psk-file PSK-file [--tls-cipher13 cipher-string] [--tls-cipher cipher-string] [-T] [-N] [-r] -i input-file
zabbix_sender -h
zabbix_sender -V

DESCRIÇÃO

zabbix_sender é um utilitário de linha de comando para enviar dados de monitoramento para o servidor ou proxy Zabbix. No servidor Zabbix um item do tipo **Zabbix trapper** deve ser criado com a chave correspondente. Observe que a entrada valores só serão aceitos de hosts especificados em **Allowed hosts** field for this item.

OPÇÕES

-c, --config config-file

Use config-file. **Zabbix sender** lê os parâmetros do servidor do arquivo de configuração agentd. Por padrão **Zabbix sender** não lê nenhum arquivo de configuração. Apenas os parâmetros **Hostname**, **ServerActive**, **SourceIP**, **TLSConnect**, **TLSCAFile**, **TLSCRLFile**, **TLSSEServerCertIssuer**, **TLSSEServerCertSubject**, **TLSSECertFile**, **TLSSEKeyFile**, **TLSPSKIdentity** e **TLSPSKFile** são suportados. Todos os endereços definidos no parâmetro de configuração **ServerActive** do agente são usados para enviar dados. Se o envio de dados em lote falhar para um endereço, os seguintes lotes não são enviados para este endereço.

-z, --zabbix-server server

Hostname ou endereço IP do servidor Zabbix. Se um host é monitorado por um proxy, o hostname do proxy ou endereço IP devem ser usados. Quando usado junto com **--config**, substitui os valores do parâmetro **ServerActive** especificado no arquivo de configuração agentd.

-p, --port port

Especifique o número da porta do servidor Zabbix em execução no servidor. O padrão é 10051. Quando usado junto com **--config**, substitui o valor de porta do parâmetro **ServerActive** especificado no arquivo de configuração agentd.

-l, --source-address IP-address

Especifica o endereço IP de origem. Quando usado junto com **--config**, substitui o parâmetro **SourceIP** especificado no arquivo de configuração agentd.

-t, --timeout seconds

Especifica o tempo limite. Intervalo válido: 1-300 segundos (padrão: 60)

-s, --host host

Especifica o nome do host ao qual o item pertence (conforme registrado no Zabbix frontend). O endereço IP do host e o nome DNS não funcionarão. Quando usado junto com **--config**, substitui o parâmetro **Hostname** especificado no arquivo de configuração agentd.

-k, --key key

Especifica a chave do item para a qual enviar o valor.

-o, --value value

Especifica o valor do item.

-i, --input-file input-file

Carrega valores do arquivo de entrada. Especifique - como **<input-file>** para ler valores da entrada padrão. Cada linha de arquivo contém espaços em branco delimitado: **<hostname> <chave> <valor>**. Cada valor deve ser especificado em sua própria linha. Cada linha deve conter 3 espaços em branco entradas delimitadas: **<hostname> <chave> <valor>**, onde "hostname" é o nome do host monitorado conforme registrado no Zabbix frontend, "key" é a chave do item de destino e "value" - o valor a ser enviado. Especifique - como **<hostname>** para usar o nome do host do arquivo de configuração do agente ou do argumento **--host**.

Um exemplo de entradas em uma linha de arquivo:

"Linux DB3" db.connections 43

O tipo de valor deve ser definido corretamente na configuração do item do Zabbix frontend. O Zabbix Sender enviará até 250 valores em uma conexão. O limite de tamanho para o envio de valores de um arquivo de entrada depende do tamanho descrito no protocolo de comunicação do Zabbix. O conteúdo do arquivo de entrada deve estar na codificação UTF-8. Todos os valores do arquivo de entrada são enviados em ordem sequencial de cima para baixo. Entradas deve ser formatado usando as seguintes regras:

- Entradas entre aspas e sem aspas são suportadas.
- Aspas duplas são os caracteres de aspas
- As entradas com espaço em branco devem ser citadas.
- Caracteres de aspas duplas e barra invertida dentro da entrada entre aspas devem ser escapadas com uma barra invertida.
- O escape não é suportado em entradas sem aspas.
- As sequências de escape de alimentação de linha (\n) são suportadas em strings entre aspas.
- As sequências de escape de alimentação de linha são cortadas a partir do final de uma entrada.

-T, --with-timestamps

Esta opção só pode ser usada com a opção **--input-file**.

Cada linha do arquivo de entrada deve conter 4 entradas delimitadas por espaços em branco: **<hostname> <key> <timestamp> <value>**. O timestamp deve ser especificado no formato de data/hora Unix. Se o item de destino tem triggers que fazem referência a ele, todos os timestamps devem estar em ordem crescente, caso contrário o cálculo do evento não estará correto.

Um exemplo de uma linha do arquivo de entrada:

"Linux DB3" db.connections 1429533600 43

Para mais detalhes, veja a opção **--input-file**.

Se um valor com timestamp for enviado para um host que está em manutenção "no data", esse valor será descartado; porém, é possível enviar um valor com timestamp para uma manutenção expirada período e será aceito.

-N, --with-ns

Esta opção só pode ser usada com a opção **--with-timestamps**.

Cada linha do arquivo de entrada deve conter 5 entradas delimitadas por espaços em branco: **<hostname> <key> <timestamp> <ns> <value>**.

Um exemplo de uma linha do arquivo de entrada:

"Linux DB3" db.connections 1429533600 7402561 43

Para mais detalhes, veja a opção **--input-file**.

-r, --real-time

Envie os valores um a um assim que forem recebidos. Isso pode ser usado ao ler da entrada padrão.

--tls-connect value

Como se conectar ao servidor ou proxy. Valores:

unencrypted

conectar sem criptografia (default)

psk

conectar usando TLS e uma chave pré-compartilhada

cert

conectar usando TLS e um certificado

--tls-ca-file CA-file

Caminho completo de um arquivo que contém os certificados de CA(s) de nível superior para verificação de certificado de peer.

--tls-crl-file CRL-file

Caminho completo de um arquivo contendo certificados revogados.

--tls-server-cert-issuer cert-issuer

Emissor de certificado de servidor permitido.

--tls-server-cert-subject cert-subject

Assunto do certificado do servidor permitido.

--tls-cert-file cert-file

Nome do caminho completo de um arquivo que contém o certificado ou cadeia de certificados.

--tls-key-file key-file

Caminho completo de um arquivo que contém a chave privada.

--tls-psk-identity PSK-identity

Cadeia de identidade PSK.

--tls-psk-file PSK-file

Caminho completo de um arquivo que contém a chave pré-compartilhada.

--tls-cipher13 cipher-string

Sequência de criptografia para OpenSSL 1.1.1 ou mais recente para TLS 1.3. Substituir o critério de seleção padrão do conjunto de cifras. Esta opção não está disponível se a versão do OpenSSL é inferior a 1.1.1.

--tls-cipher cipher-string

String de prioridade GnuTLS (para TLS 1.2 e superior) ou string de cifra OpenSSL (somente para TLS 1.2). Substitua os critérios de seleção padrão do ciphersuite.

-v, --verbose

Modo detalhado, **-vv** para mais detalhes.

-h, --help

Exibir esta ajuda e sair.

-V, --version

Informações de versão e sair.

STATUS DE SAÍDA

O status de saída é 0 se os valores foram enviados e todos eles foram processado com sucesso pelo servidor. Se os dados foram enviados, mas o processamento de pelo menos um dos valores falhou, o status de saída é 2. Se o envio de dados falhou, o status de saída é 1.

EXEMPLOS

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -k mysql.queries -o 342.45

Envie **342.45** como o valor para o item **mysql.queries** do host monitorado. Use host monitorado e Zabbix Server definido no arquivo de configuração do Zabbix Agent.

zabbix_sender -c /etc/zabbix/zabbix_agentd.conf -s "Monitored Host" -k mysql.queries -o 342.45

Envie **342.45** como o valor para o item **mysql.queries** do host **Monitored Host** usando o Zabbix Server definido no arquivo de configuração do Zabbix Agent.

zabbix_sender -z 192.168.1.113 -i data_values.txt

Envie valores do arquivo **data_values.txt** para o Zabbix Server com IP **192.168.1.113**. Nomes de host e keys são definidos no arquivo.

echo "- hw.serial.number 1287872261 SQ4321ASDF" | zabbix_sender -c /usr/local/etc/zabbix_agentd.conf -T -i -

Envie um valor de timestamp da linha de comando para o Zabbix Server, especificado no arquivo de configuração do Zabbix Agent. O traço (hífen) nos dados de entrada indica que o nome do host também deve ser usado no mesmo arquivo de configuração.

echo "'Zabbix server' trapper.item ''" | zabbix_sender -z 192.168.1.113 -p 10000 -i -

Enviar valor vazio de um item para o Zabbix Server com endereço IP **192.168.1.113** na porta **10000** da linha de comando. Valores vazios deve ser indicado por aspas duplas vazias.

zabbix_sender -z 192.168.1.113 -s "Monitored Host" -k mysql.queries -o 342.45 --tls-connect cert --tls-ca-file /home/zabbix/zabbix_ca_file --tls-cert-file /home/zabbix/zabbix_agentd.crt --tls-key-file /home/zabbix/zabbix_agentd.key

Envie **342.45** como o valor para o item **mysql.queries** no host **Monitored Host** para servidor com IP **192.168.1.113** usando TLS com certificado.

zabbix_sender -z 192.168.1.113 -s "Monitored Host" -k mysql.queries -o 342.45 --tls-connect psk --tls-psk-identity "PSK ID Zabbix agentd" --tls-psk-file /home/zabbix/zabbix_agentd.psk

Envie **342.45** como o valor para o item **mysql.queries** no host **Monitored Host** para servidor com IP **192.168.1.113** usando TLS com chave pré-compartilhada (PSK).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), zabbix_get(1), zabbix_proxy(8), zabbix_server(8), zabbix_js(1), zabbix_agent2(8), zabbix_web_service(8)

Autor

Alexei Vladishev <alex@zabbix.com>

Index

NOME

SINÓPSE

DESCRIÇÃO

OPÇÕES

STATUS DE SAÍDA

EXEMPLOS

VEJA TAMBÉM

AUTOR

Este documento foi criado em: 11 de junho de 2021, às 08:42:39 GMT

zabbix_server

Seção: Comandos de Manutenção (8)

Atualizado: 2020-09-04

[Índice](#) [Retornar ao conteúdo principal](#)

NOME

zabbix_sender - servidor Zabbix.

SINOPSE

zabbix_server [-c config-file]

zabbix_server [-c config-file] **-R** runtime-option

zabbix_server **-h**

zabbix_server **-V**

DESCRIÇÃO

zabbix_server é o daemon principal do software Zabbix.

OPÇÕES

-c, --config config-file

Use o config-file alternativo em vez do padrão.

-f, --foreground

Execute o servidor Zabbix em primeiro plano.

-R, --runtime-control runtime-option

Execute funções administrativas de acordo com a opção de tempo de execução.

-h, --help Mostrar essa ajuda e saia.

-V, --version Exiba informação de versão e saia.

Exemplos de Servidor Zabbix executando com parâmetros de linha de comando:

```
shell> zabbix_server -c /usr/local/etc/zabbix_server.conf
shell> zabbix_server --help
shell> zabbix_server -V
```

CONTROLE DE TEMPO DE EXECUÇÃO

Opções de controle de tempo de execução

config_cache_reload

Recarregue o cache de configuração. Ignorado se o cache estiver sendo carregado no momento. O arquivo de configuração padrão (a menos que a opção **-c** seja especificada) será usado para encontrar o arquivo PID e o sinal será enviado ao processo, listado no arquivo PID.

snmp_cache_reload

Recarrega o cache SNMP.

housekeeper_execute

Executa o housekeeper. Ignorado se o housekeeper já estiver sendo executado. **trigger_housekeeper_execute** inicia o procedimento trigger housekeeping. Ignorado se o procedimento trigger housekeeping estiver em progresso no momento.

diaginfo[=section]

Registre as informações de diagnóstico interno da seção especificada. Seção pode ser historycache, preprocessing, alerting, lld, valuecache. Por padrão, as informações de diagnóstico de todas as seções são registradas.

ha_status

Registre o estado do cluster de alta disponibilidade (HA).

ha_remove_node[=target]

Remove o nó de alta disponibilidade especificado por seu nome ou ID. Note que nós ativos/em espera não podem ser removidos.

ha_set_failover_delay[=delay]

Define o atraso de failover de alta disponibilidade. Sufixos de tempo são suportados, ex. 10s, 1m.

secrets_reload

Recarrega secrets do Cofre.

service_cache_reload

Recarrega o cache do gerenciador de serviço.

snmp_cache_reload

Recarrega o cache SNMP, limpa as propriedades SNMP (engine time, engine boots, engine id, credenciais) para todos os hosts.

prof_enable[=target]

Habilita profiling. Afeta todos os processos se o destino não é especificado. Profiling habilitado provê detalhes de todos rwlocks/mutexes por nome da função. Suportado desde o Zabbix 6.0.13.

prof_disable[=target]

Desabilitar profiling. Afeta todos os processos se o destino não é especificado. Suportado desde o Zabbix 6.0.13.

log_level_increase[=target]

Aumenta o nível de detalhamento do log, afeta todos os processos se o destino não é especificado

log_level_decrease[=target]

Diminui o nível de detalhamento do log, afeta todos os processos se o destino não é especificado

Metas de controle de nível de registro

process-type

Todos os processos do tipo especificado (alerter, alert manager, configuration syncer, discoverer, escalator, history syncer, housekeeper, http poller, icmp pinger, ipmi manager, ipmi poller, java poller, lld manager, lld worker, poller, preprocessing manager, preprocessing worker, proxy poller, self-monitoring, snmp trapper, task manager, timer, trapper, unreachable poller, vmware collector)

process-type,N

Tipo e número do processo (e.g., poller,3)

pid

Identificador de processo, até 65535. Para valores maiores, especifique o destino como "tipo de processo, N"

ARQUIVOS

/usr/local/etc/zabbix_server.conf.

Localização padrão do arquivo de configuração do Zabbix Server (se não for modificado durante a compilação).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), **zabbix_get**(1), **zabbix_proxy**(8), **zabbix_sender**(1), **zabbix_js**(1), **zabbix_agent2**(8)

AUTOR

Alexei Vladishev <alex@zabbix.com>

Index

[NOME](#)

[SINÓPSE](#)

[DESCRIÇÃO](#)

[OPÇÕES](#)

[ARQUIVOS](#)

[VEJA TAMBÉM](#)

[AUTOR](#)

Este documento foi criado em: 04 de setembro de 2020, às 16:12:14 GMT

zabbix_web_service

Seção: Comandos para manutenção(8)

Atualizado: 2019-01-29

[Index](#) [Voltar ao conteúdo principal](#)

NOME

zabbix_web_service - Zabbix web service

SINOPSE

zabbix_web_service [-c config-file]

zabbix_web_service -h

zabbix_web_service -V

DESCRIÇÃO

zabbix_web_service é um aplicação para fornecer serviços web para componentes Zabbix.

OPÇÕES

-c, --config config-file

Use o config-file alternativo ao invés do padrão. **-h, --help**

Exibir esta ajuda e sair.

-V, --version

Saída de informações de versões e saída.

ARQUIVOS

/usr/local/etc/zabbix_web_service.conf

Localização padrão do arquivo de configuração do serviço web Zabbix (se não modificado durante o tempo de compilação).

VEJA TAMBÉM

Documentação <https://www.zabbix.com/manuals>

zabbix_agentd(8), **zabbix_get**(1), **zabbix_proxy**(8), **zabbix_sender**(1), **zabbix_server**(8), **zabbix_js**(1), **zabbix_agent2**(8)

AUTOR

Zabbix LLC

Índice

NOME

SINOPSE

DESCRIÇÃO

OPÇÕES

ARQUIVOS

VEJA TAMBÉM

AUTOR

Este documento foi criado em: 11 de junho de 2021, às 12:58:30 GMT